**BAB IV**

**PERANCANGAN SISTEM**

* 1. **Perancangan Sistem**

Perancangan sistem adalah sebuah kegiatan merancang dan menentukan cara mengolah sistem informasi dari hasil analisa sistem sehingga dapat memenuhi kebutuhan dari pengguna termasuk diantaranya perancangan user interface, data dan aktivitas proses (O’Brien dan Marakas 2009: 639).

* + 1. **Identifikasi Aktor**

Pada Sistem Informasi *Manajemen* & *Marketing* Usaha Makaroni Masim ini terdapat 3 pengguna atau aktor. Adapun penjelasan dari pengguna atau aktor adalah sebagai berikut :

**Tabel 4.1** Identifikasi Aktor

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1 | Pegawai (Administrator) | Pengguna yang memiliki hak akses untuk mengelola *website* secara keseluruhan. Pengguna ini memiliki hak akses untuk mengelola setiap data master, seperti data produk, data pelanggan, data stok, data transaksi, data produk masuk, data produk keluar dan kelola laporan perusahaan. |
| 2 | Pemilik | Pengguna yang mengelola data pegawai, *request* keseluruhan laporan perusahaan, seperti laporan keuangan,laporan produk keluar,laporan produk masuk dan laporan pemesanan produk. |
| 3 | Pelanggan | Pengguna yang membeli produk di toko Musim Masim Bandung |

* + 1. **Identifikasi *Use Case***

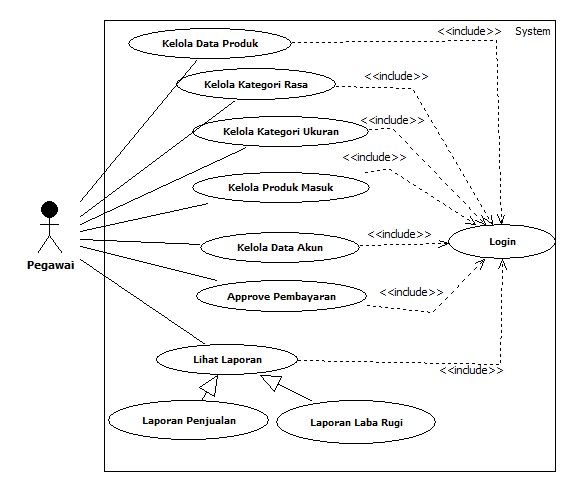
Pada Sistem Informasi *Manajemen* & *Marketing* Usaha Makaroni Masim ini terdapat beberapa *use case* untuk mengetahui fungsi dalam sistem yang akan dibangun Adapun penjelasan dari *use case* tersebut adalah sebagai berikut:

**Tabel 4.2** Identifikasi *Use Case*

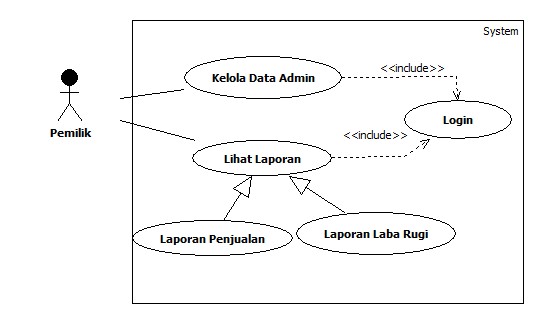
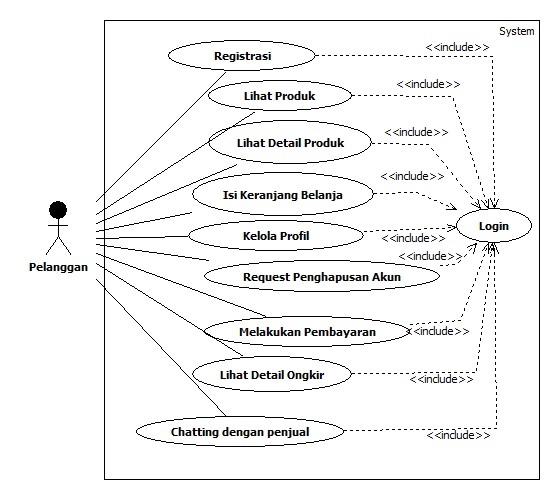
|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1 | Login | [Pemilik, Pegawai, Pelanggan] Aktivitas yang mengatur hak akses pengguna dan merupakan validasi pengguna untuk mendapatkan hak akses ke sistem |
| 2 | Kelola Data Produk | [Pegawai] Aktivitas untuk mengelola data produk baik itu aksi tambah, edit dan hapus produk. |
| 3 | Kelola Kategori Rasa | [Pegawai] Aktivitas untuk mengelola data rasa produk Makaroni Masim seperti aksi tambah, edit dan hapus. |
| 4 | Kelola Kategori Ukuran | [Pegawai] Aktivitas untuk mengelola data ukuran produk Makaroni Masim seperti aksi tambah, edit dan hapus. |
| 5 | Kelola Data Admin | [Pemilik] Aktivitas untuk mengelola data pegawai/Admin |
| 6 | Kelola Data Akun | [Pegawai/Admin] Aktivitas untuk mengelola data akun pembeli seperti hapus akun |
| 7 | Kelola Produk Masuk | [Pegawai] Aktivitas untuk mengelola data produk masuk ketika terjadi penambahan stok produk |
| 8 | Approve Pembayaran | [Pegawai] Aktivitas Approve pembayaran oleh pegawai ketika pembeli sudah berhasil melakukan pembayaran |
| 9 | Lihat Laporan | [Pemilik, Pegawai] Aktivitas untuk melihat laporan baik itu laporan penjualan dan laporan laba rugi. |
| 10 | Registrasi | [Pelanggan] Aktivitas Registrasi oleh pelanggan yang sebelumnya belum memiliki akun. |
| 11 | Lihat Produk | [Pelanggan] Aktivitas untuk melihat produk |
| 12 | Lihat Detail Produk | [Pelanggan] Aktivitas untuk melihat detail produk |
| 13 | Isi Keranjang Belanja | [Pelanggan] Proses pengisian keranjang setelah pemilihan produk yang diinginkan |
| 14 | Chatting dengan penjual | [Pelanggan] Aktivitas chatting langsung via WA dengan penjual |
| 15 | Lihat Detail Ongkir | [Pelanggan] Proses melihat ongkos kirim, estimasi waktu dan jenis jasa pengiriman produk |
| 16 | Request Penghapusan Akun | [Pelanggan] Aktivitas Request penghapusan Akun kepada pihak admin |
| 17 | Kelola Profil | [Pelanggan] Aktivitas edit profil |
| 18 | Melakukan Pembayaran | [Pelanggan] Aktivitas pembayaran yang bisa dilakukan via M-Banking, Indomart, Alfamart atau ATM dll jenis Pembayaran Online |

**4.2 Metode Perancangan Sistem**

Metode perancangan sistem untuk Sistem Informasi Manajemen & Marketing ini diusulkan akan berorientasi objek dengan menggunakan notasi *Unified Modeling Language* (UML) dalam bentuk *Use Case* Diagram dan *Activity* Diagram sebagai berikut :

* + 1.  ***Use Case Diagram***

**Gambar 4.1** *Use Case* Pegawai

******Gambar 4.2** *Use Case* Pelanggan

**Gambar 4.3** *Use Case* Pemilik

***Use Case Spesification***

1. *Use Case Spesification Login*

**Tabel 4.3** *Use Case Spesification* Login

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Login | |
| *Actor* | Pemilik Perusahaan, Pegawai dan Pelanggan | |
| *Description* | Sistem mengautentikasi *login* aktor dengan meng-*query* tabel setiap aktor melalui inputan *username* dan *password*. Tujuan dari *use case* ini adalah setiap aktor dapat memasuki sistem sesuai hak aksesnya masing-masing. | |
| *Pre-condition* | Sistem menampilkan *interface* dan *form login* | |
| *Basic Flow* | Aktor | Sistem |
| 1. Menginputkan *username* dan *password* | 1. Sistem akan memvalidasi *username* dan *password* yang diinputkan *user* |
| *Alternative Flow* | Jika *username* dan *password* salah maka sistem menampilkan pesan “*Username dan Password Salah*”  Jika *username* dan *password* kosong maka sistem menampilkan pesan “*Username dan Password Tidak Boleh Kosong*” | |
| *Requirement* | Memiliki hak akses untuk *login* | |
| *Post -condition* | Sistem menampilkan halaman utama | |

1. *Use Case Spesification* Kelola Data Produk

**Tabel 4.4** *Use Case Spesification* Kelola Data Produk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola Data Produk | |
| *Actor* | Pegawai | |
| *Description* | *Use case* ini digunakan oleh pegawai untuk mengelola data produk. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, edit dan hapus data produk. | |
| *Pre-condition* | *User* dalam keadaan login | |
| *Basic Flow* | Aktor | Sistem |
| 1. Memilih tombol data produk | 1. Menampilkan halaman data produk |
| ***A. Tambah***   1. Mengklik *link* tambah data | 1. Menampilkan form tambah data |
| 1. Mengisi *form* tambah data, kemudian menekan tombol *save* | 1. Memvalidasi data inputan, jika data lengkap dan sesuai system menyimpan data ke *database,* menampilkan data ke datagrid dan menampilkan notifikasi “Data berhasil disimpan!”. (*alternatif flow* 6.1) |
| ***B. Ubah***   1. Mengklik *link* ubah datapada *action* data grid | 1. Menampilkan *form* ubah yang berisi detail data yang akan diubah |
| 1. Mengubah data, kemudian menekan tombol *save* | 1. Memvalidasi data ubah, menyimpan data ke *database,* menampilkan data ke *datagrid*, menampilkan notifikasi “Data berhasil disimpan!” (*alternatif flow* 6.1) |
| ***C.Hapus***  *3.* Mengklik *link* hapuspada *action* data grid | 1. Menampilkan konfirmasi hapus data |
| 1. Mengklik tombol hapus pada konfirmasi hapus data (*alternatif flow* 5.1) | 1. Menghapus data, menampilkan notifikasi “Data Telah Terhapus” |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pegawai | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (tambah, ubah dan hapus data) | |

1. *Use Case Spesification* Kelola Kategori Rasa

**Tabel 4.5** *Use Case Spesification* Kelola Kategori Rasa

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola Kategori Rasa | |
| *Actor* | Pegawai | |
| *Description* | *Use case* ini digunakan oleh pegawai untuk mengelola data kategori rasa. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, edit dan hapus data rasa. | |
| *Pre-condition* | *User* dalam keadaan login | |
| *Basic Flow* | Aktor | Sistem |
| 1. Memilih tombol data rasa | 1. Menampilkan halaman data rasa |
| ***A. Tambah***   1. Mengklik *link* tambah data | 1. Menampilkan form tambah data |
| 1. Mengisi *form* tambah data, kemudian menekan tombol *save* | 1. Memvalidasi data inputan, jika data lengkap dan sesuai sistem menyimpan data ke *database,* menampilkan data ke *datagrid* dan menampilkan notifikasi “Data berhasil disimpan!”. (*alternatif flow* 6.1) |
| ***B. Ubah***   1. Mengklik *link* ubah datapada *action* datagrid | 1. Menampilkan *form* ubah yang berisi detail data yang akan diubah |
| 1. Mengubah data, kemudian menekan tombol *save* | 1. Memvalidasi data ubah, menyimpan data ke *database,* menampilkan data ke *datagrid*, menampilkan notifikasi “Data berhasil disimpan!” (*alternatif flow* 6.1) |
| ***C.Hapus***   1. Mengklik *link* hapuspada *action* data grid | 1. Menampilkan konfirmasi hapus data |
| 1. Mengklik tombol hapus pada konfirmasi hapus data (*alternatif flow* 5.1) | 1. Menghapus data, menampilkan notifikasi “Data Telah Terhapus” |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pegawai | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (tambah, ubah dan hapus data) | |

1. *Use Case Spesification* Kelola Kategori Ukuran

**Tabel 4.6** *Use Case Spesification* Kelola Kategori Ukuran

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola Kategori Ukuran | |
| *Actor* | Pegawai | |
| *Description* | *Use case* ini digunakan oleh pegawai untuk mengelola data kategori ukuran produk. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, edit dan hapus data ukuran produk. | |
| *Pre-condition* | *User* dalam keadaan login | |
| *Basic Flow* | Aktor | Sistem |
| 1. Memilih tombol data ukuran | 1. Menampilkan halaman data ukuran |
| ***A. Tambah***   1. Mengklik *link* tambah data | 1. Menampilkan form tambah data |
| 1. Mengisi *form* tambah data, kemudian menekan tombol *save* | 1. Memvalidasi data inputan, jika data lengkap dan sesuai sistem menyimpan data ke *database,* menampilkan data ke *datagrid* dan menampilkan notifikasi “Data berhasil disimpan!”. (*alternatif flow* 6.1) |
| ***B. Ubah***   1. Mengklik *link* ubah datapada *action* datagrid | 1. Menampilkan *form* ubah yang berisi detail data yang akan diubah |
| 1. Mengubah data, kemudian menekan tombol *save* | 1. Memvalidasi data ubah, menyimpan data ke *database,* menampilkan data ke *datagrid*, menampilkan notifikasi “Data berhasil disimpan!” (*alternatif flow* 6.1) |
| ***C.Hapus***  *3.* Mengklik *link* hapuspada *action* data grid | 1. Menampilkan konfirmasi hapus data |
| 1. Mengklik tombol hapus pada konfirmasi hapus data (*alternatif flow* 5.1) | 1. Menghapus data, menampilkan notifikasi “Data Telah Terhapus” |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pegawai | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (tambah, ubah dan hapus data) | |

1. *Use Case Spesification* Kelola Data Admin

**Tabel 4.7** *Use Case Spesification* Kelola Data Admin

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola Data Admin | |
| *Actor* | Pemilik Perusahaan | |
| *Description* | *Use case* ini digunakan oleh pemilik perusahaan untuk mengelola data admin atau pegawai. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, edit dan hapus data admin. | |
| *Pre-condition* | *User* dalam keadaan login | |
| *Basic Flow* | Aktor | Sistem |
| 1. Memilih tombol data admin | 1. Menampilkan halaman data admin |
| ***A. Tambah***   1. Mengklik *link* tambah data | 1. Menampilkan form tambah data |
| 1. Mngisi *form* tambah data, kemudian menekan tombol *save* | 1. Memvalidasi data inputan, jika data lengkap dan sesuai sistem menyimpan data ke *database,* menampilkan data ke *datagrid* dan menampilkan notifikasi “Data berhasil disimpan!”. (*alternatif flow* 6.1) |
| ***B. Ubah***   1. Mengklik *link* ubah datapada *action* datagrid | 1. Menampilkan *form* ubah yang berisi detail data yang akan diubah |
| 1. Mengubah data, kemudian menekan tombol *save* | 1. Memvalidasi data ubah, menyimpan data ke *database,* menampilkan data ke *datagrid*, menampilkan notifikasi “Data berhasil disimpan!” (*alternatif flow* 6.1) |
| ***C.Hapus***  *3.* Mengklik *link* hapuspada *action* data grid | 1. Menampilkan konfirmasi hapus data |
| 1. Mengklik tombol hapus pada konfirmasi hapus data (*alternatif flow* 5.1) | 1. Menghapus data, menampilkan notifikasi “Data Telah Terhapus” |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pemilik perusahaan | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (tambah, ubah dan hapus data) | |

1. *Use Case Spesification* Kelola Data Pembeli

**Tabel 4.8** *Use Case Spesification* Kelola Data Pembeli

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola Data Pembeli | |
| *Actor* | Pegawai | |
| *Description* | *Use case* ini digunakan oleh pegawai perusahaan untuk mengelola data pembeli. Fungsi-fungsi yang ada di dalam *use case* ini yaitu hapus data pembeli jika *action request false*. | |
| *Pre-condition* | *User* dalam keadaan login | |
| *Basic Flow* | Aktor | Sistem |
| 1. Memilih tombol data admin | 1. Menampilkan halaman data admin |
| ***A. Hapus***  *3.* Mengklik *link* hapuspada *action* data grid | 1. Menampilkan konfirmasi hapus data |
| 1. Mengklik tombol hapus pada konfirmasi hapus data (*alternatif flow* 5.1) | 1. Menghapus data, menampilkan notifikasi “Data Telah Terhapus” |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pegawai | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* ( hapus data) | |

1. *Use Case Spesification* Kelola Produk Masuk

**Tabel 4.9** *Use Case Spesification* Kelola Produk Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola Produk Masuk | |
| *Actor* | Pegawai | |
| *Description* | *Use case* ini digunakan oleh pegawai perusahaan untuk mengelola data admin atau pegawai. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, edit dan hapus data produk masuk. | |
| *Pre-condition* | *User* dalam keadaan login | |
| *Basic Flow* | Aktor | Sistem |
| 1. Memilih tombol data produk masuk | 1. Menampilkan halaman data produk masuk |
| ***A. Tambah***   1. Mengklik *link* tambah data | 1. Menampilkan form tambah data |
| 1. Mngisi *form* tambah data, kemudian menekan tombol *save* | 1. Memvalidasi data inputan, jika data lengkap dan sesuai sistem menyimpan data ke *database,* menampilkan data ke *datagrid* dan menampilkan notifikasi “Data berhasil disimpan!”. (*alternatif flow* 6.1) |
| ***B. Ubah***   1. Mengklik *link* ubah datapada *action* datagrid | 1. Menampilkan *form* ubah yang berisi detail data yang akan diubah |
| 1. Mengubah data, kemudian menekan tombol *save* | 1. Memvalidasi data ubah, menyimpan data ke *database,* menampilkan data ke *datagrid*, menampilkan notifikasi “Data berhasil disimpan!” (*alternatif flow* 6.1) |
| ***C.Hapus***  *3.* Mengklik *link* hapuspada *action* data grid | 1. Menampilkan konfirmasi hapus data |
| 1. Mengklik tombol hapus pada konfirmasi hapus data (*alternatif flow* 5.1) | 1. Menghapus data, menampilkan notifikasi “Data Telah Terhapus” |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pegawai | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (tambah, ubah dan hapus data) | |

1. *Use Case Spesification Approve* Pembayaran

**Tabel 4.10** *Use Case Spesification Approve* Pembayaran

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Approve* Pembayaran | |
| *Actor* | Pegawai | |
| *Description* | *Use case* ini digunakan oleh pegawai perusahaan untuk konfirmasi pembayaran. Fungsi-fungsi yang ada di dalam *use case* ini yaitu *confirm* pembayaran. | |
| *Pre-condition* | *User* dalam keadaan login | |
| *Basic Flow* | Aktor | Sistem |
| 1. Memilih tombol transaksi pembayaran | 1. Menampilkan halaman pembayaran |
| ***A.Confirm***  *3.* Mengklik *link* *confirm* pada *action* data grid | 1. Menampilkan konfirmasi *confirm* pembayaran |
| 1. Mengklik tombol *confirm* (*alternatif flow* 5.1) | 1. Menyimpan data transaksi penjualan ke *database*, menampilkan notifikasi “Pembayaran berhasil dikonfirmasi” |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2) | |
| *Requirement* | Berhasil login sebagai pegawai | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (tambah data) | |

1. *Use Case Spesification* Melihat Laporan

**Tabel 4.11** *Use Case Spesification* Melihat Laporan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *Use Case Name* | Melihat Laporan | | | | | |
| *Actor* | Pemilik Perusahaan, Pegawai | | | | | |
| *Description* | Pemilik perusahaan dan pegawai dapat melihat dan mencetak laporan penjualan dan laporan laba rugi | | | | | |
| *Precondition* | Sistem menampilkan form utama *view* data laporan yang berisi form *view* data laporan | | | | | |
| *Basic Flow* | Aktor | | | Sistem | | |
| 1. Klik tombol laporan | *view* | data | 2. Menampilkan |  | laporan |
| 3. ***Cetak,*** Mengklik tombol cetak pada *datagrid* tabel | | | 4. Menampilkan form tanggal awal dan tanggal akhir data yang akan di cetak | | |
|  | 5. Mengisi tanggal dan menekan tombol simpan | | | 6. Menampilkan data yang akan dicetak | | |
| *Alternative Flow* | - | | | | | |
| *Requirement* | Berhasil *login* sebagai pemilik perusahaan dan pegawai | | | | | |
| *Post Condition* | Sistem menampilkan data laporan | | | | | |
| *Relationship* | *Generalization Relationship :*  Melihat Laporan Penjualan dan Laporan Laba Rugi | | | | | |

1. *Use Case Spesification* Registrasi

**Tabel 4.12** *Use Case Spesification* Registrasi

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Registrasi | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk membuat akun pelanggan guna untuk login ke sistem | |
| *Pre-condition* | *User* belum memiliki akun | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* membuka *website* untuk pertama kali atau pengguna baru | 1. Menampilkan halaman login |
| 1. *User* mengklik *link registrasi* | 1. Manampilkan form inputan registrasi |
| 1. Mngisi *form* inputan data, kemudian menekan tombol *save* | 1. Memvalidasi data inputan, jika data lengkap dan sesuai sistem menyimpan data ke *database,* menampilkan *form* login kembali dan menampilkan notifikasi “Registrasi berhasil!”. (*alternatif flow* 6.1) |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pemilik perusahaan | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (tambah data pelanggan) | |

1. *Use Case Spesification* Lihat Produk

**Tabel 4.13** *Use Case Spesification* Lihat Produk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Lihat Produk | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk melihat produk | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik *dropdown shop* di *navbar* | 1. Menampilkan pilihan menu *dropdown* (lihat produk,cart dan checkout) |
| 1. *User* memilih lihat produk | 1. Manampilkan halaman berisi produk-produk |
| *Alternative Flow* | - | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Data produk berhasil ditampilkan | |

1. *Use Case Spesification* Lihat Detail Produk

**Tabel 4.14** *Use Case Spesification* Lihat Detail Produk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Lihat Detail Produk | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk melihat detail produk | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik *dropdown shop* di *navbar* | 1. Menampilkan pilihan menu *dropdown* (lihat produk,cart dan checkout) |
| 1. *User* mengklik memilih lihat produk | 1. Manampilkan halaman berisi produk-produk |
|  | 1. *User* mengklik *icon* detail produk | 1. Menampilkan halaman berisi detail produk |
| *Alternative Flow* | - | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Data Detail produk berhasil ditampilkan | |

1. *Use Case Spesification* Isi Keranjang Belanja

**Tabel 4.15** *Use Case Spesification* Isi Keranjang Belanja

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Isi Keranjang Belanja | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk proses pengisian keranjang belanja | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik *dropdown shop* di *navbar* | 1. Menampilkan pilihan menu *dropdown* (lihat produk,cart dan checkout) |
| 1. *User* mengklik memilih lihat produk | 1. Manampilkan halaman berisi produk-produk |
| 1. *User* mengklik tombol *add to cart* | 1. Menampilkan halaman berisi detail produk |
| 1. *User* memilih kategori ukuran produk dan kemudian mengklik tombol *add to cart* | 1. Menampilkan notifikasi “Berhasil ditambahkan ke keranjang!”, dan menyimpan data ke *database.* |
| *Alternative Flow* | - | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Berhasil menyimpan produk ke dalam keranjang belanja | |

1. *Use Case Spesification Chatting* dengan penjual

**Tabel 4.16** *Use Case Spesification Chatting* dengan penjual

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Chatting* dengan penjual | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk *chatting* dengan penjual | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik *link contact us* di *navbar* | 1. Menampilkan halaman *chatting* dengan penjual |
| 1. *User* mengklik *button Chatting via WhataApp* | 1. Mengarahkan ke halaman *chatting via WhatsApp* |
| *Alternative Flow* | - | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Berhasil diarahkan ke halaman *chatting WhatsApp* | |

1. *Use Case Spesification* Lihat Detail Ongkir

**Tabel 4.17** *Use Case Spesification* Lihat Detail Ongkir

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Lihat Detail Ongkir | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk melihat detail ongkos kirim produk baik itu biaya ongkir, jenis jasa pengiriman,jenis layanan pengiriman dan estimasi waktu pengiriman | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik *dropdown cart* di *navbar* | 1. Menampilkan halaman berisi produk di *cart* |
| 1. *User* memilih produk yang ingin dilihat detail ongkir nya, kemudian *click button checkout* dibawah halaman | 1. Manampilkan halaman berisi inputan kota asal, kota tujuan dan jasa pengiriman.   Berat produk terisi secara otomatis |
| 1. *User* menginputkan dengan cara memilih kota asal, kota tujuan dan jenis jasa pengiriman yang diinginkan, kemudian mengklik *button* cek ongkir | 1. Menampilkan detail ongkos kirim seperti biaya pengiriman, jenis jasa pengiriman, jenis layanan pengiriman dan estimasi waktu pemgiriman |
| *Alternative Flow* | - | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Berhasil menampilkan detail ongkir | |

1. *Use Case Spesification Request* Penghapusan Akun

**Tabel 4.18** *Use Case Spesification Request* Penghapusan Akun

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Request* Penghapusan Akun | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk proses *Request* penghapusan akun | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik tombol *my account* di *navbar* | 1. Menampilkan halaman yang berisi detail *account* |
| 1. *User* mengklik tombol ajukan penghapusan akun ke admin | 1. Mangirimkan *request* penghapusan akun kepada admin, dan mengubah status akun pembeli menjadi *false* |
| *Alternative Flow* | - | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Request penghapusan akun terkirim ke admin untuk segera dihapus | |

1. *Use Case Spesification* Kelola *Profile*

**Tabel 4.18** *Use Case Spesification* Kelola *Profile*

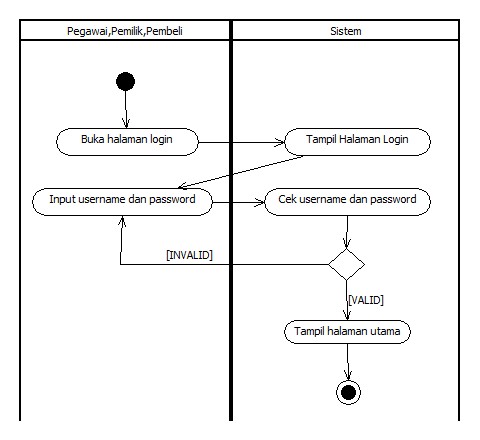
|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola *Profile* | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk proses edit *profile* | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik tombol *my account* di *navbar* | 1. Menampilkan halaman yang berisi detail *account* |
| 1. *User* mengklik tombol edit akun login & security | 1. Menampilkan form inputan yang akan diedit |
|  | 1. Mengedit data, dan menekan *save* | 1. Memvalidasi data yang diubah, menyimpan data ke *database,* menampilkan data ke halaman *my account* (*alternatif flow* 6.1) |
| *Alternative Flow* | 5.1 Mengklik tombol *Cancel* (kembali ke langkah 2)  6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 4) | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Sistem melakukan *Data Manipulation Language* (DML) pada database sesuai dengan aksi yang dilakukan *actor* (edit data pelanggan) | |

1. *Use Case Spesification* Melakukan Pembayaran

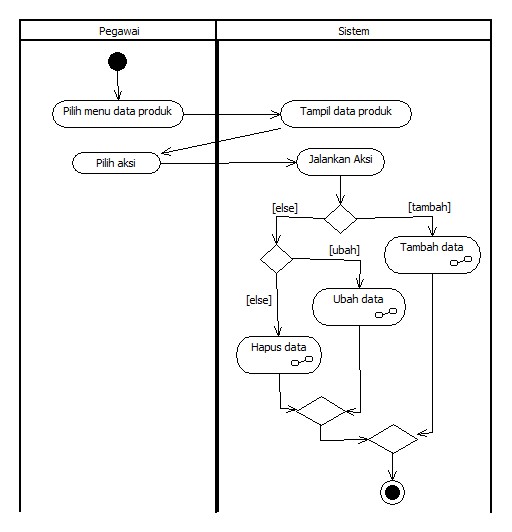
**Tabel 4.20** *Use Case Spesification* Melakukan Pembayaran

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Kelola *Profile* | |
| *Actor* | Pelanggan | |
| *Description* | *Use case* ini digunakan oleh pelanggan untuk proses edit *profile* | |
| *Pre-condition* | *User* telah login | |
| *Basic Flow* | Aktor | Sistem |
| 1. *User* mengklik *link* pilih metode pembayaran | 1. Menampilkan *pop up* berisi menu jenis metode pembayaran |
| 1. *User* memilih jenis pembayaran, dan menekan tombol buat pesanan | 1. Menampilkan halaman pembayaran berisi kode pembayaran atau rekening tujuan dan menunggu konfirmasi dari admin |
| *Alternative Flow* | - | |
| *Requirement* | Berhasil login sebagai pelanggan | |
| *Post-condition* | Sistem berhasil melakukan proses pembayaran | |

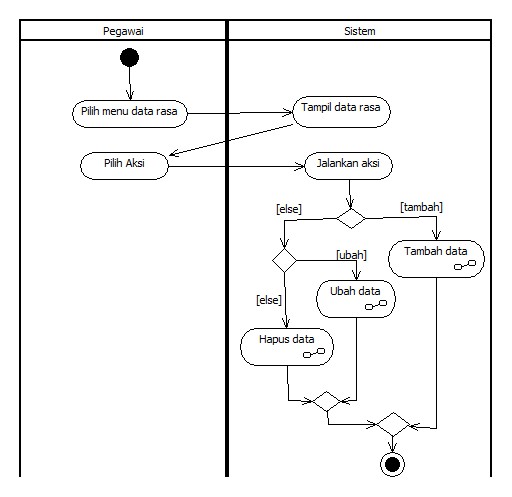
* + 1. **Activity Diagram**

1. *Activity Diagram* Login

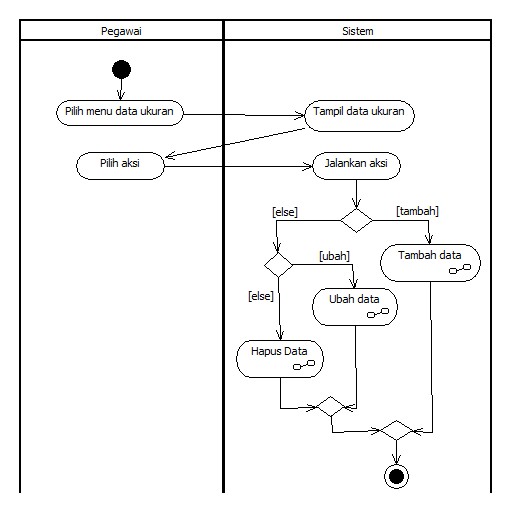
**Gambar 4.4** *Activity Diagram* Login

1. *Activity Diagram* Kelola Data Produk

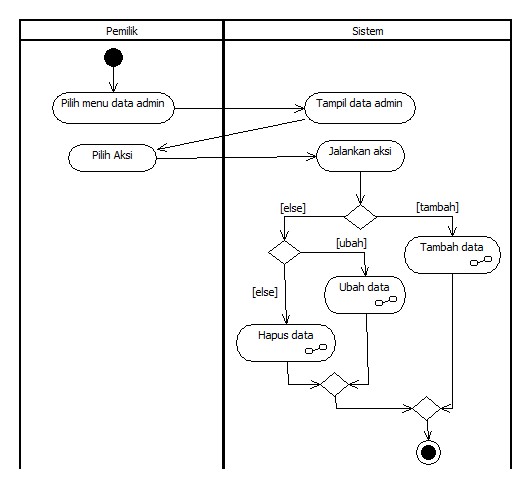
**Gambar 4.5** *Activity Diagram* Kelola Data Produk

1. *Activity Diagram* Kelola Kategori Rasa

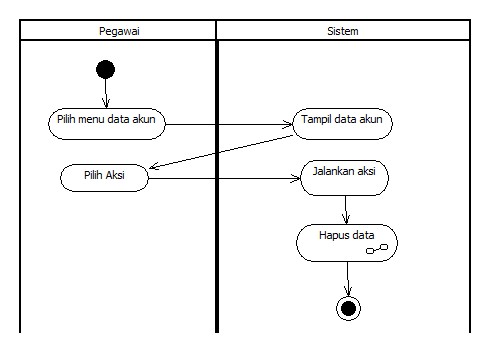
**Gambar 4.6** *Activity Diagram* Kelola Kategori Rasa

1. *Activity Diagram* Kelola Kategori Ukuran

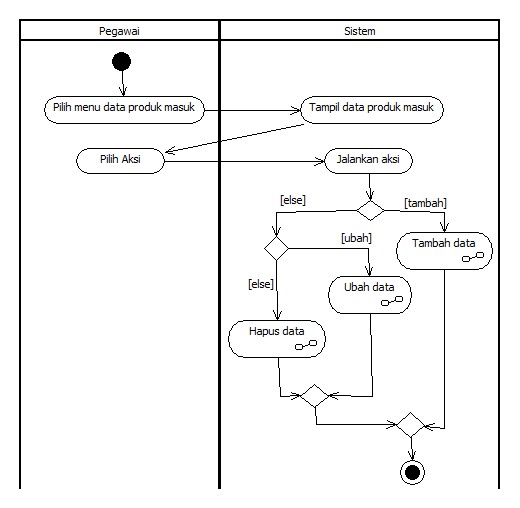
**Gambar 4.7** *Activity Diagram* Kelola Kategori Ukuran

1. *Activity Diagram* Kelola Data Admin

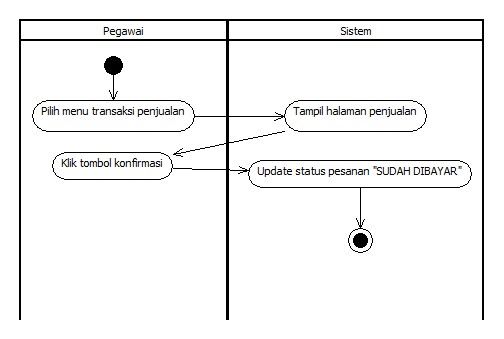
**Gambar 4.8** *Activity Diagram* Kelola Data Admin

1. *Activity Diagram* Kelola Data Akun

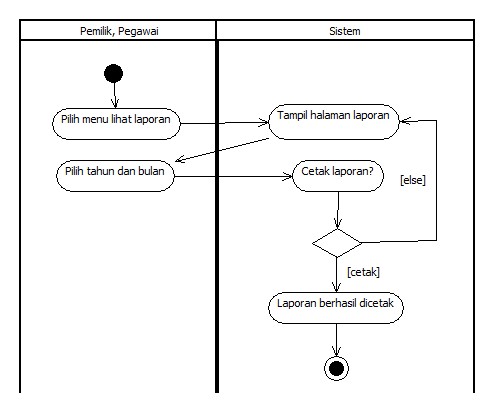
**Gambar 4.9** *Activity Diagram*  Kelola Data Akun

1. *Activity Diagram* Kelola Data Produk Masuk

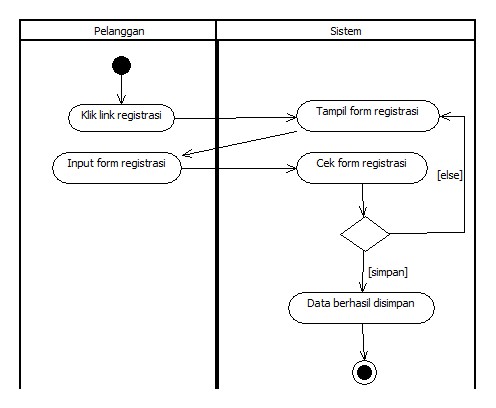
**Gambar 4.10** *Activity Diagram* Kelola Data Produk Masuk

1. *Activity Diagram Approve* Pembayaran

**Gambar 4.11** *Activity Diagram Approve* Pembayaran

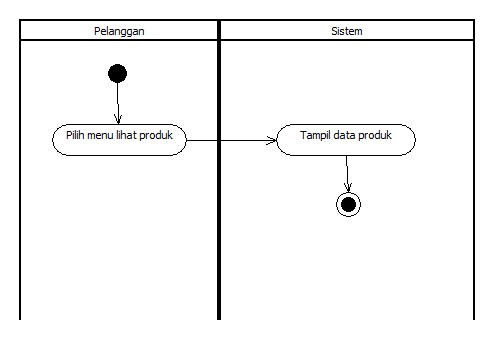
1. *Activity Diagram* Lihat Laporan

**Gambar 4.12** *Activity Diagram* Lihat Laporan

1. *Activity Diagram* Registrasi

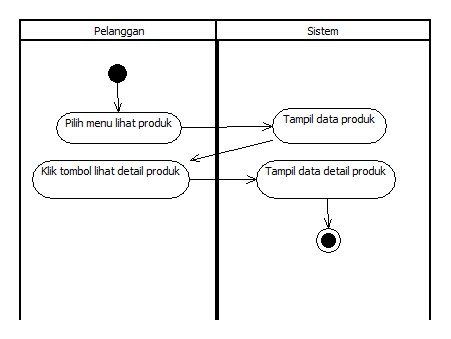
**Gambar 4.13** *Activity Diagram* Registrasi

1. *Activity Diagram* Lihat Produk



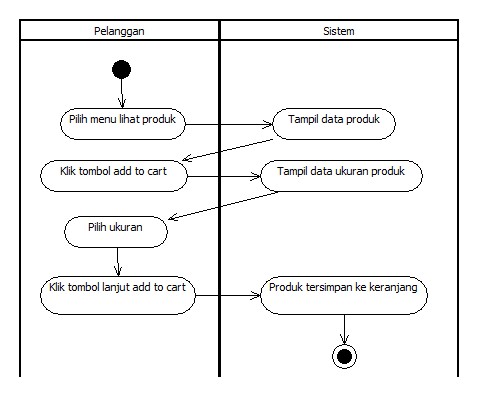
**Gambar 4.14** *Activity Diagram* Lihat Produk

1. *Activity Diagram* Lihat Detail Produk



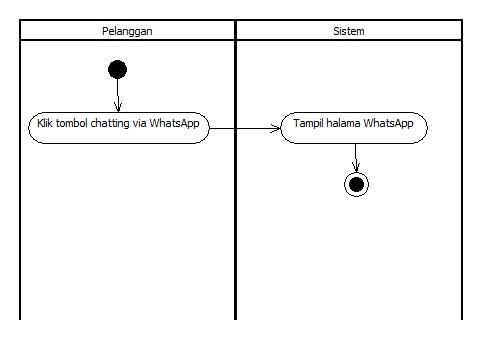
**Gambar 4.15** *Activity Diagram* Lihat Detail Produk

1. *Activity Diagram* Isi Keranjang Belanja



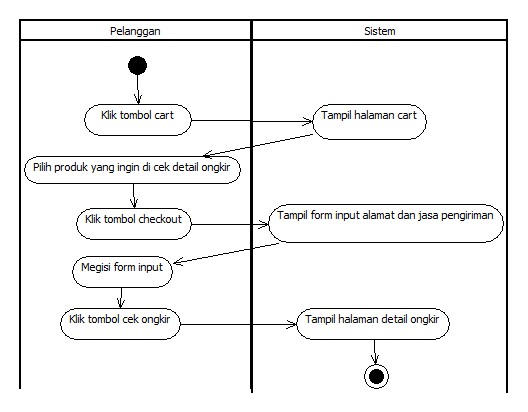
**Gambar 4.16** *Activity Diagram* Isi Keranjang Belanja

1. *Activity Diagram Chatting* dengan penjual

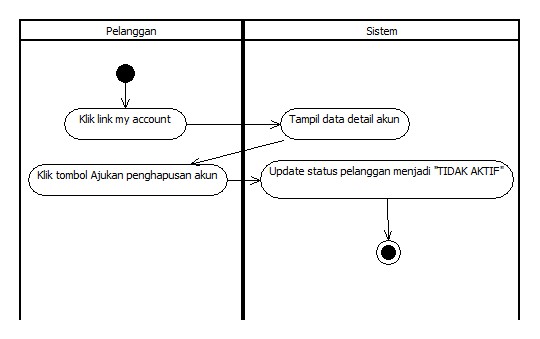


**Gambar 4.17** *Activity Diagram* Isi Keranjang Belanja

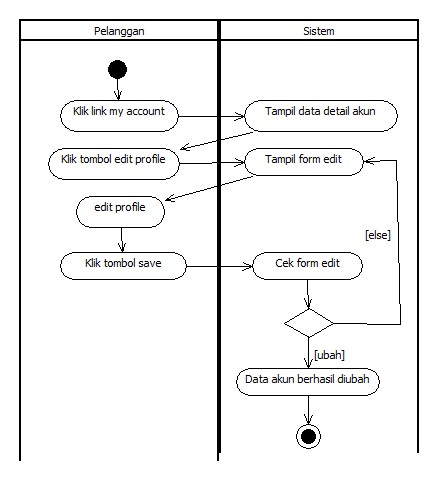
1. *Activity Diagram* Lihat Detail Ongkir



**Gambar 4.18** *Activity Diagram* Lihat Detail Ongkir

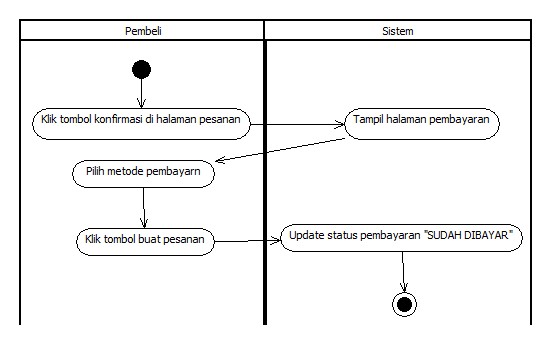
1. *Activity Diagram Request* Penghapusan Akun

**Gambar 4.19** *Activity Diagram Request* Penghapusan Akun

1. *Activity Diagram* Kelola *Profile*

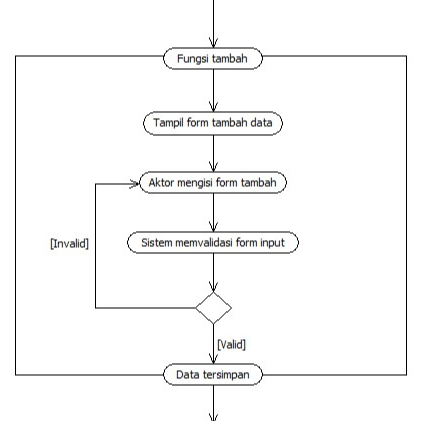
**Gambar 4.20** *Activity Diagram* Kelola *Profile*

1. *Activity Diagram* Kelola Pembayaran



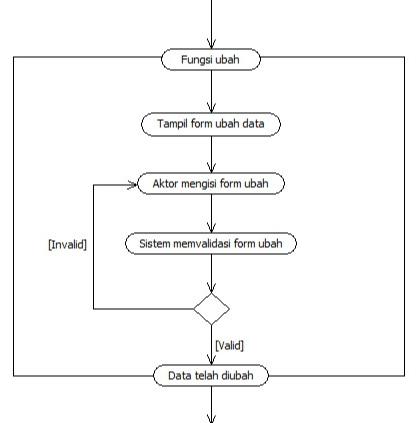
**Gambar 4.21** *Activity Diagram* Kelola Pembayaran

* *Sub-Activity* jalankan fungsi tambah



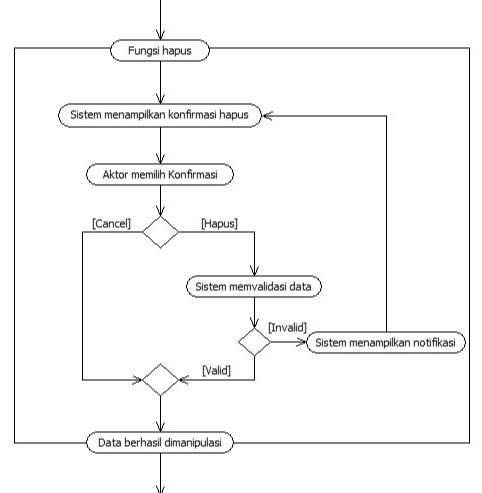
**Gambar 4.22** *Sub-Activity* Fungsi Tambah

* *Sub-Activity* jalankan fungsi ubah



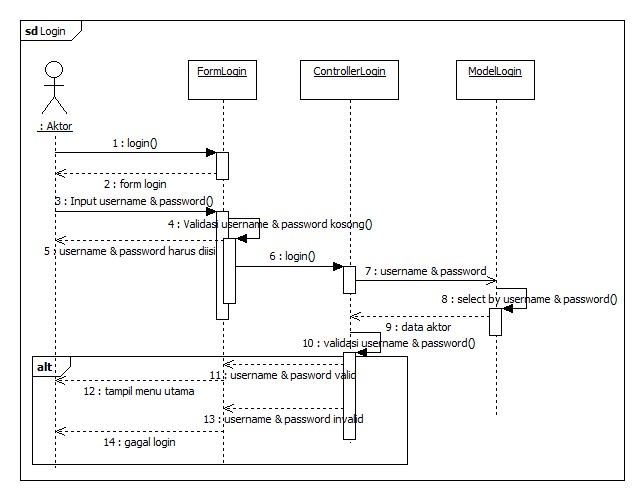
**Gambar 4.23** *Sub-Activity* Fungsi ubah

* *Sub-Activity* jalankan fungsi hapus

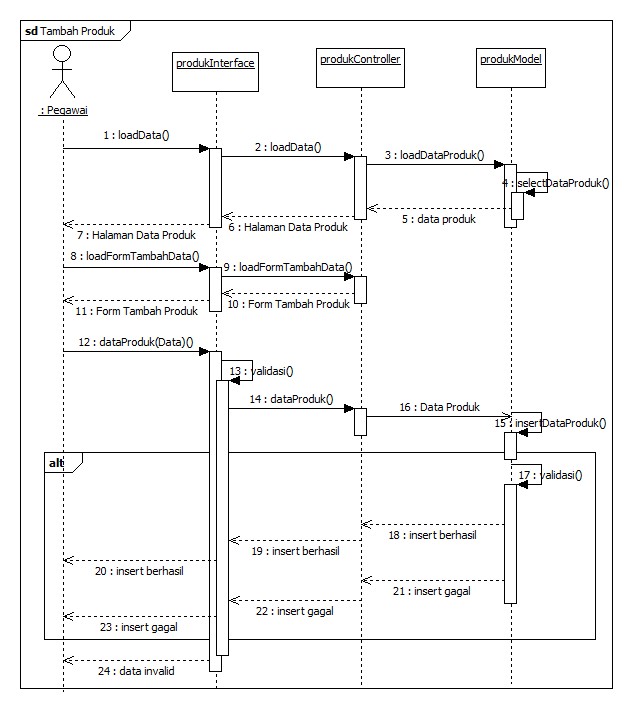


**Gambar 4.24** *Sub-Activity* Fungsi hapus

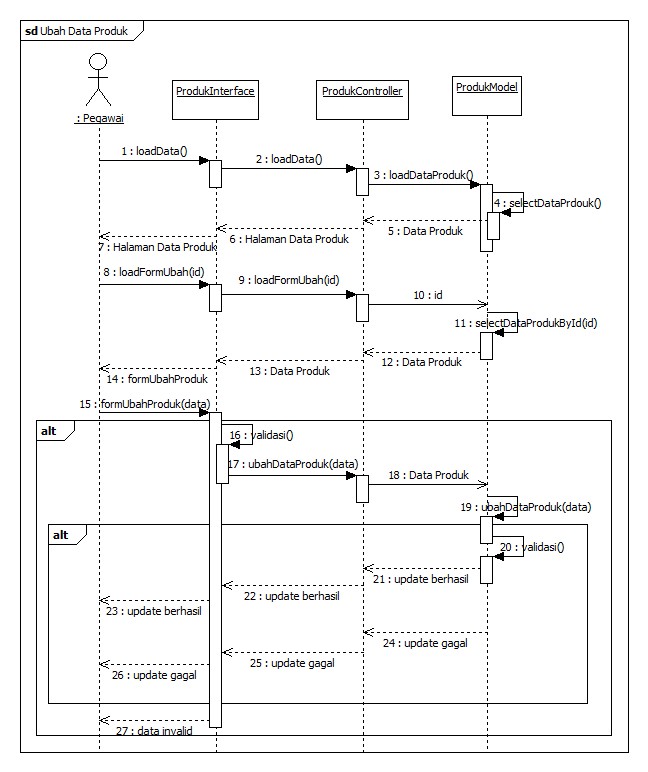
* + 1. ***Sequence Diagram***

1. *****Sequence Diagram* Login Aktor

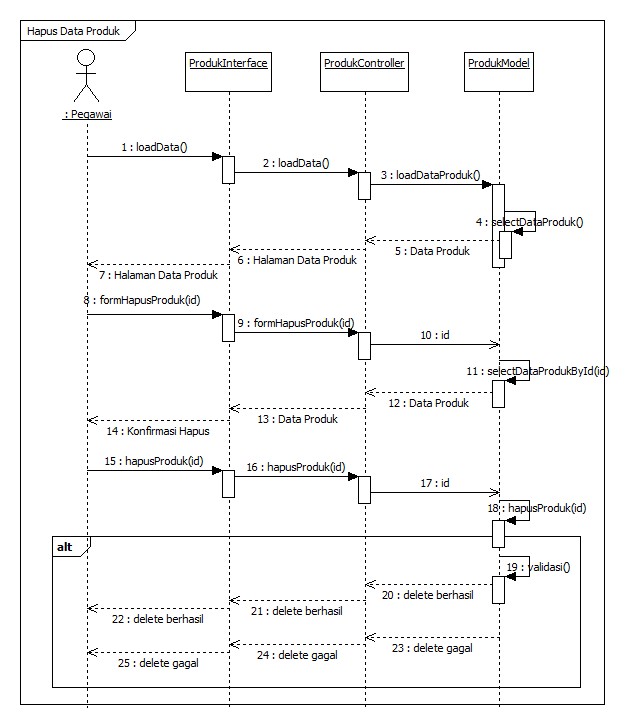
**Gambar 4.25** *Sequence* Diagram Login Pegawai

1. *Sequence Diagram* Tambah Produk

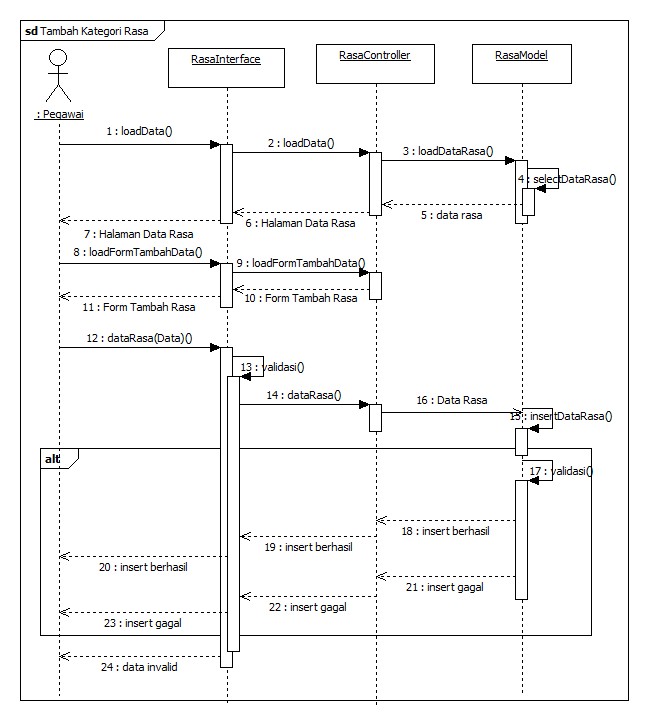
**Gambar 4.26** *Sequence Diagram* Tambah Produk

1. *Sequence Diagram* Ubah Data Produk

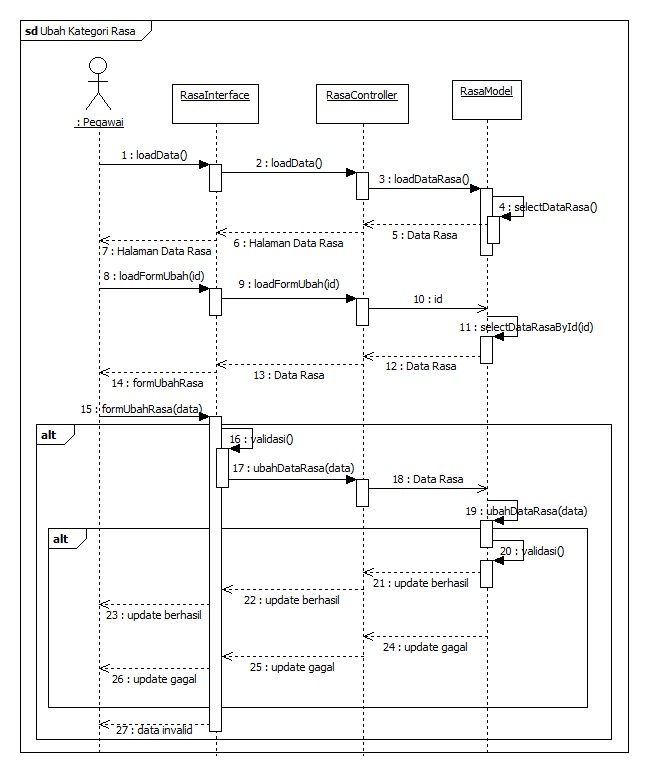
**Gambar 4.27** *Sequence Diagram* Ubah Data Produk

1. *Sequence Diagram* Hapus Data Produk

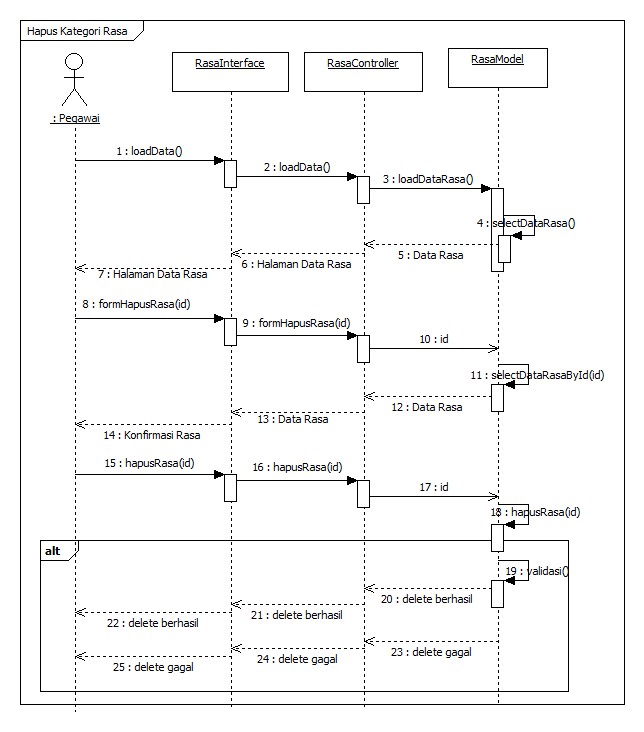
**Gambar 4.28** *Sequence Diagram* Hapus Data Produk

1. *****Sequence Diagram* Tambah Kategori Rasa

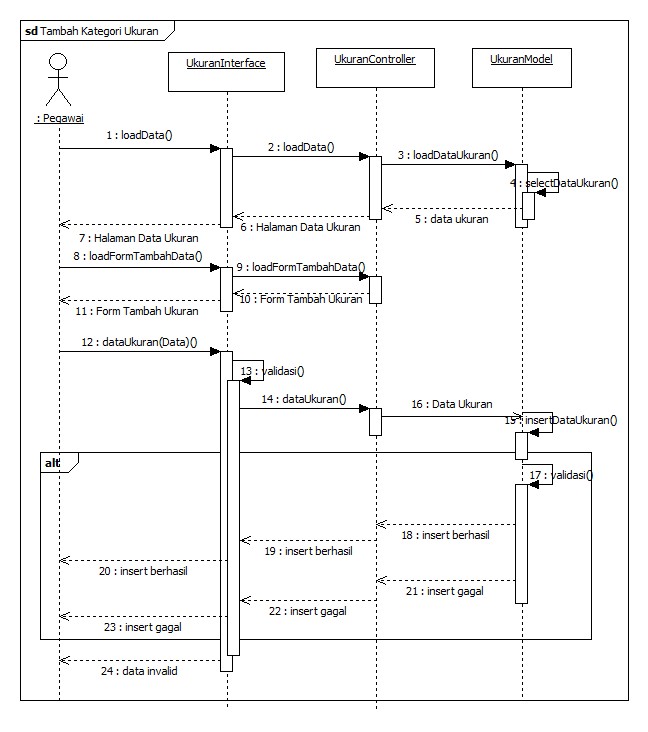
**Gambar 4.29** *Sequence Diagram* Tambah Kategori Rasa

1. *Sequence Diagram* Ubah Kategori Rasa

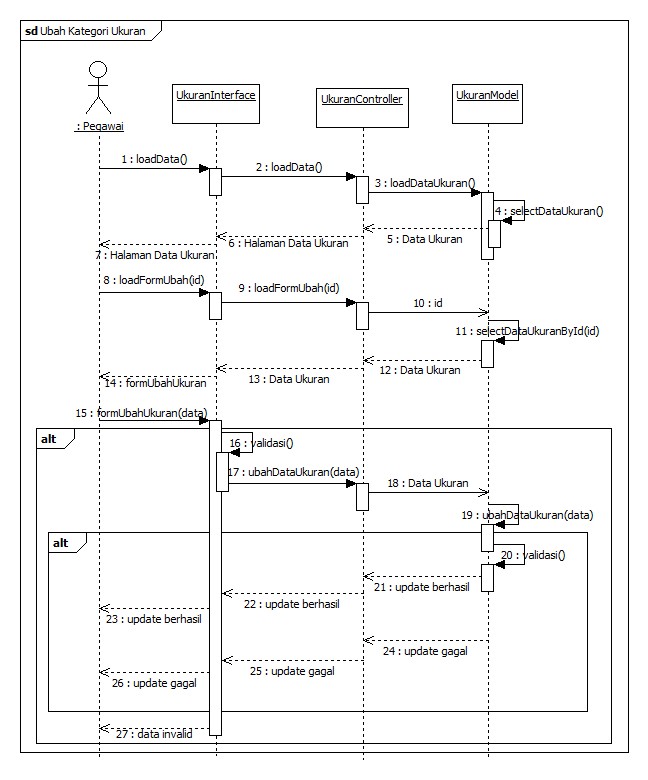
**Gambar 4.30** *Sequence Diagram* Ubah Kategori Rasa

1. *Sequence Diagram* Hapus Kategori Rasa

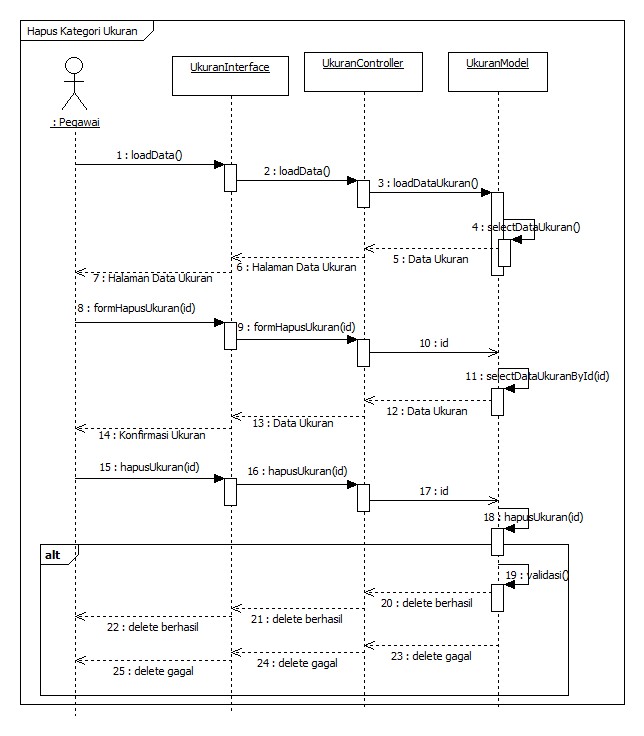
**Gambar 4.31** *Sequence Diagram* Hapus Kategori Rasa

1. *****Sequence Diagram* Tambah Kategori Ukuran

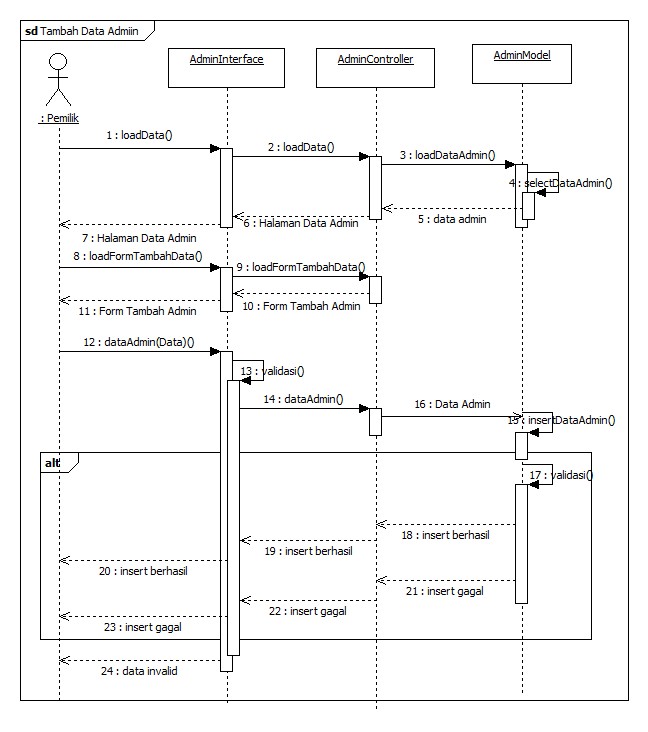
**Gambar 4.32** *Sequence Diagram* Tambah Kategori Ukuran

1. *Sequence Diagram* Ubah Kategori Ukuran

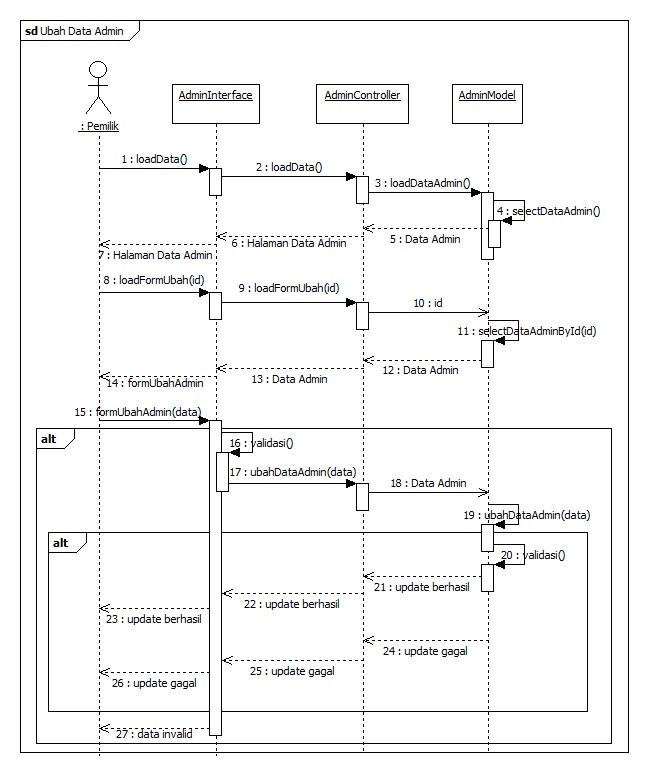
**Gambar 4.33** *Sequence Diagram* Ubah Kategori Ukuran

1. *Sequence Diagram* Hapus Kategori Ukuran

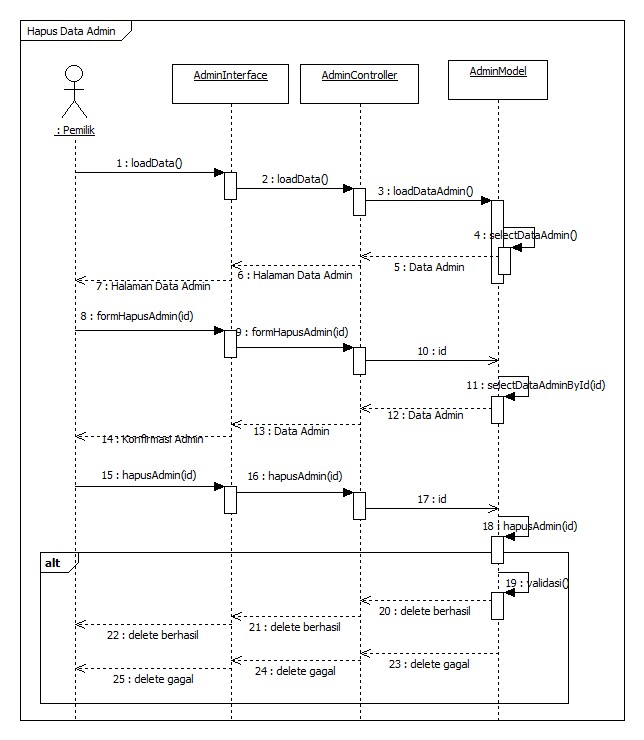
**Gambar 4.34** *Sequence Diagram* Hapus Kategori Ukuran

1. *****Sequence Diagram* Tambah Data Admin

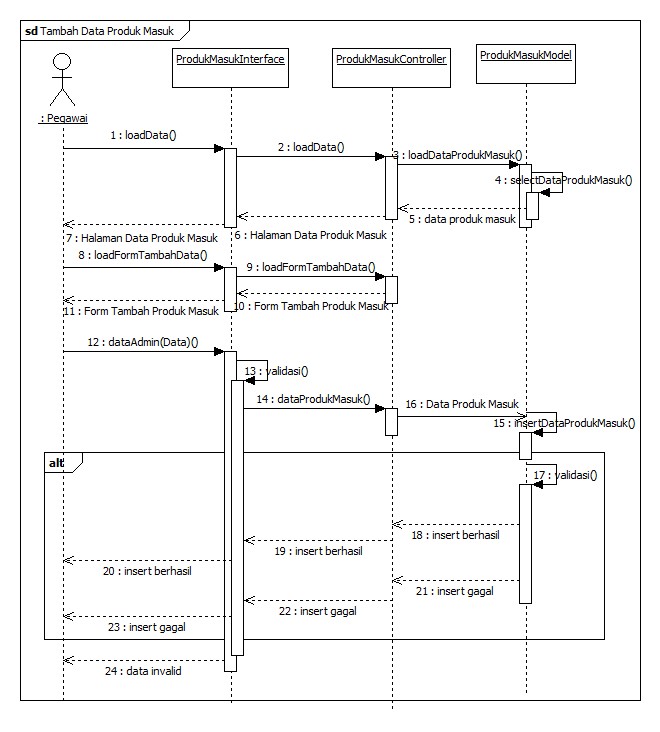
**Gambar 4.35** *Sequence Diagram* Tambah Data Admin

1. *Sequence Diagram* Ubah Data Admin

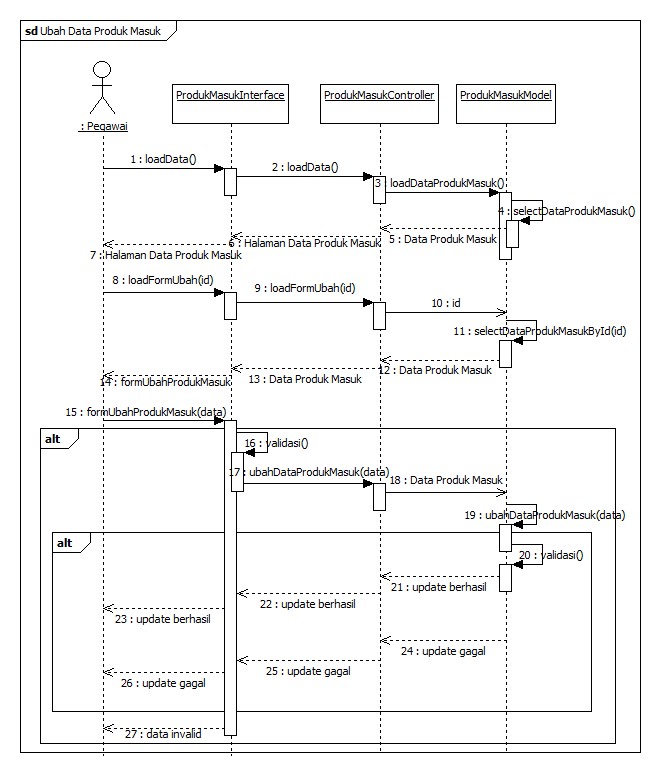
**Gambar 4.36** *Sequence Diagram* Ubah Data Admin

1. *Sequence Diagram* Hapus Data Admin

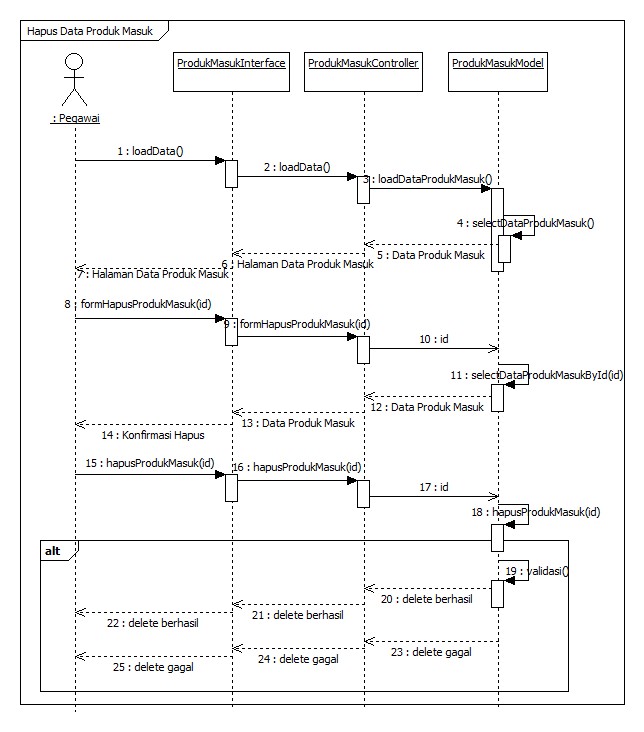
**Gambar 4.37** *Sequence Diagram* Hapus Data Admin

1. *Sequence Diagram* Tambah Data Produk Masuk

**Gambar 4.38** *Sequence Diagram* Tambah Data Produk Masuk

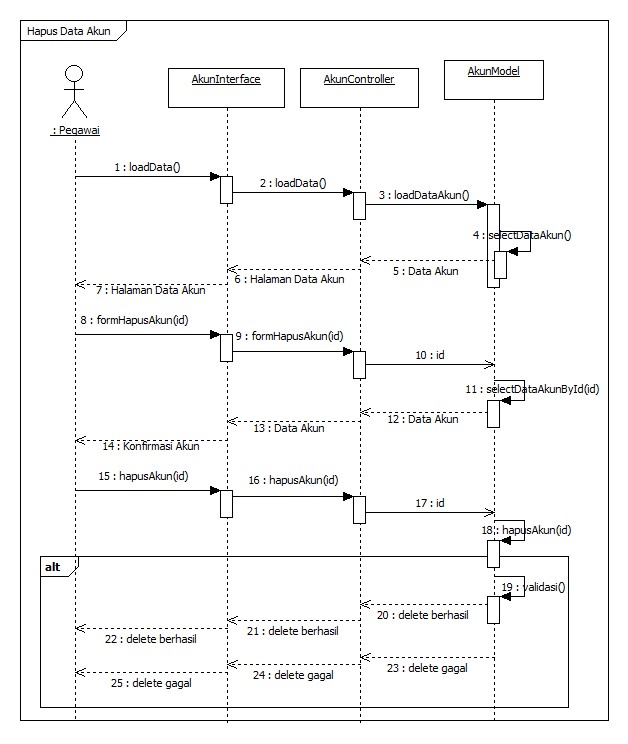
1. *Sequence Diagram* Ubah Data Produk Masuk

**Gambar 4.39** *Sequence Diagram* Ubah Data Produk Masuk

1. *Sequence Diagram* Hapus Data Produk Masuk

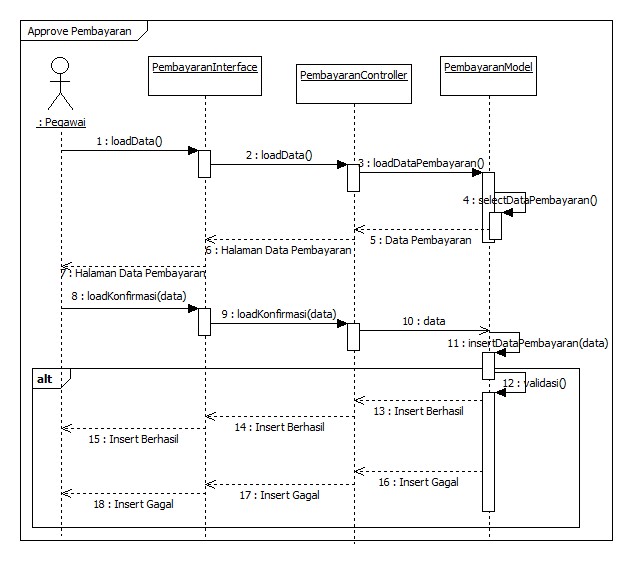
**Gambar 4.40** *Sequence Diagram* Hapus Data Produk Masuk

1. *Sequence Diagram* Hapus Data Akun

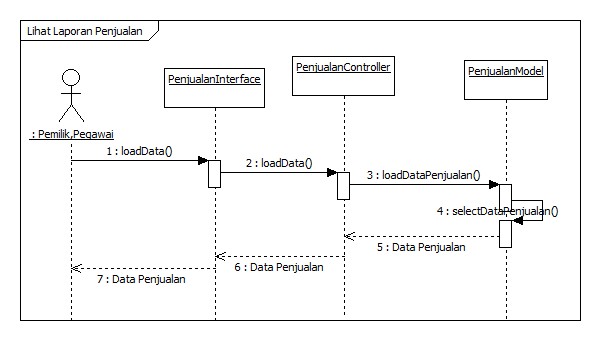


**Gambar 4.41** *Sequence Diagram* Hapus Data Akun

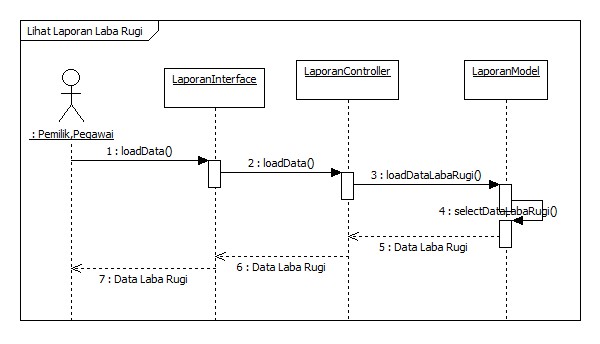
1. *Sequence Diagram Approve* Pembayaran



**Gambar 4.42** *Sequence Diagram* *Approve* Pembayaran

1. *Sequence Diagram* Lihat Laporan Penjualan

**Gambar 4.43** *Sequence Diagram* Lihat Laporan Penjualan

1. *Sequence Diagram* Lihat Laporan Laba Rugi

**Gambar 4.44** *Sequence Diagram* Lihat Laporan Laba Rugi