**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***

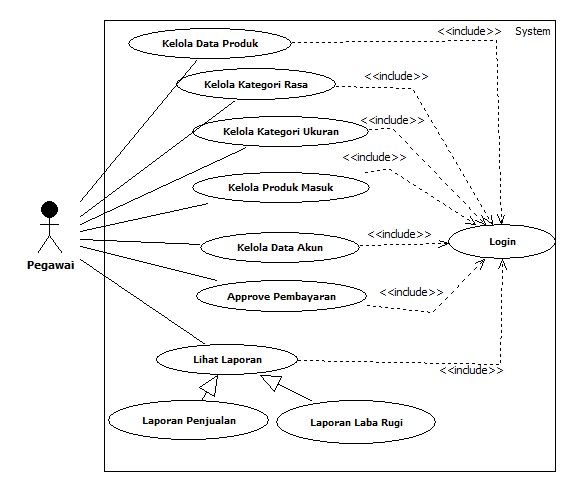
Pada Sistem Informasi *Manajemen* & *Marketing* Usaha Makaroni Masim ini terdapat beberapa *use case* untuk mengetahui fungsi dalam sistem yang akan dibangun Adapun penjelasan dari *use case* tersebut adalah sebagai berikut:

**Tabel 4.2** Identifikasi *Use Case*

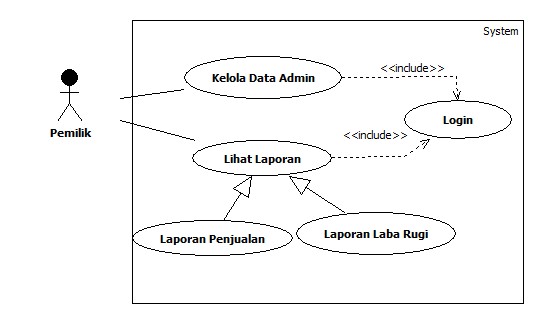
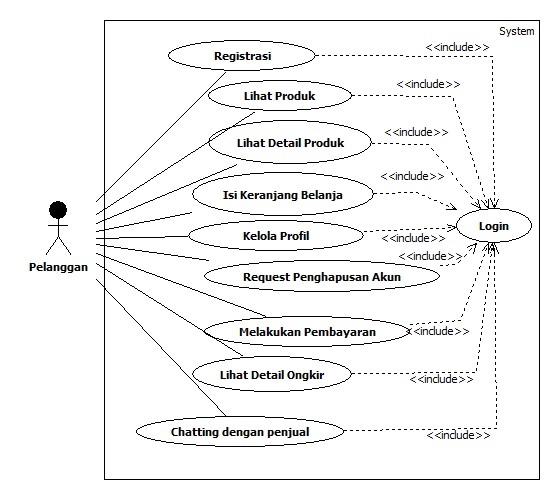
|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1 | Login | [Pemilik, Pegawai, Pelanggan] Aktivitas yang mengatur hak akses pengguna dan merupakan validasi pengguna untuk mendapatkan hak akses ke sistem |
| 2 | Kelola Data Produk | [Pegawai] Aktivitas untuk mengelola data produk 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 penjualan | 1. Menampilkan halaman penjualan produk |
| ***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)  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 data) | |

1. *Use Case Spesification* Melihat Laporan

**Tabel 4.10** *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 pada *datagrid* | data | 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 4.7** *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) | |

**Tabel 4.5** *Use Case Spesification* Kelola Transaksi Penjualan

|  |  |  |
| --- | --- | --- |
| Use Case Name | Kelola Transaksi Penjualan | |
| Aktor | Pegawai | |
| Description | *Use Case* ini digunakan oleh pihak pegawai untuk mengelola transaksi penjualan Makaroni Masim | |
| Precondition | Sistem menampilkan notifikasi terjadinya pemesanan produk dari pelanggan | |
| Basic Flow | Aktor | Sistem |
| 1. Klik tombol kelola Transaksi Penjualan | 1. Sistem menampilkan form tambah transaksi penjualan |
| 1. Mengisi data transaksi baru pada form transaksi penjualan, kemudian menekan tombol save | 1. Memvalidasi data inputan, menyimpan data ke *database,* menampilkan data ke datagrid, menampilkan notifikasi “Data Transaski Tersimpan”. (*alternatif flow* 6.1) |
| Alternative Flow | 6.1 Data yang diinput tidak sesuai dengan *type* data yang diprogram dalam sistem (kembali ke langkah 2) | |
| Requirement | Berhasil login sesuai hak akses aktor masing-masing | |
| Post Condition | Sistem melakukan *Data manipulation Languange (DML)*pada table produk masuk di database penjualan sesuai input actor dan menampilkan hasil DML tersebut langsung di datagrid tabel pegawai (pada form aplikasi) | |

**Tabel 4.6** *Use Case Spesification* Kelola Produk Masuk

|  |  |  |
| --- | --- | --- |
| Use Case Name | Kelola Produk Masuk | |
| Aktor | Pegawai | |
| Description | *Use Case* ini digunakan oleh pihak pegawai untuk mengelola produk masuk di toko Makaroni Masim | |
| Precondition | *User* sudah berhasil masuk ke sistem | |
| Basic Flow | Aktor | Sistem |
| 1. *Klik*  tombol kelola produk masuk | 1. Menampilkan datagrid tabel produk masuk |
| 1. Meng *Klik button* insert | 1. Menampilkan form tambah data produk masuk |
|  | 1. Mengisi data produk baru pada form tambah data produk masuk, menekan tombol *Save* | 1. Memvalidasi data inputan, menyimpandata produk baru pada tabel produk masuk di database dan menampilkan nya pada *datagrid* tabel pegawai serta menampilkan pesan “data produk berhasil di simpan” (alternative 6.1) |
| Alternative Flow | * 1. data yang diisikan tidak sesuai dengan type data yang diprogram dalam sistem (kembali ke langkah 4) | |
| Requirement | Berhasil login sebagai pegawai | |
| Post Condition | Sistem memberikan notifikasi keberhasilan *login* kemudian menampilkan form utama | |

**Tabel 4.7** *Use Case Spesification* Kelola Ongkir

|  |  |  |
| --- | --- | --- |
| Use Case Name | Kelola Ongkir | |
| Aktor | Pegawai | |
| Description | *Use Case* digunakan oleh Pegawai untuk mengidentifikasi jumlah ongkir saat *checkout* produk | |
| Precondition | Sistem menampilkan notifikasi pemesanan dari pelanggan | |
| Basic Flow | Aktor | Sistem |
| 1. *Klik* tombol kelola ongkir | 1. Menampilkan form inputan daerah dan berat produk |
| 1. Mengisi form inputan daerah dan berat produk untuk menentukan jumlah ongkir sesuai dengan kerjasama dengan ekspedisi sebelumnya, menekan tombol *Save* | 1. Sistem menampilkan jumlah ongkir sesuai dengan ekspedisi yang dipilih dan memberikan notifikasi jumlah ongkir ke pelanggan |
| Alternative Flow |  | |
| Requirement | Berhasil login sebagai pegawai | |
| Post Condition | Sistem menampilkan jumlah ongkir dan memberikan notifikasi keberhasilan *cek* ongkir | |

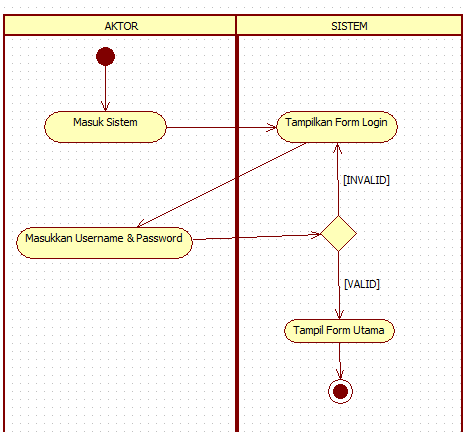
**Tabel 4.10** *Use Case Spesification* Melihat laporan

|  |  |  |
| --- | --- | --- |
| Use Case Name | Melihat laporan | |
| Aktor | Pemilik | |
| Description | Pemilik dapat melihat laporan persediaan, laporan produk keluar, laporan produk masuk dan laporan keuangan | |
| Precondition | Sistem menampilkan form utama *view* data laporan yang berisi form *view data* laporan | |
| Basic Flow | Aktor | Sistem |
| 1. Klik tombol *view* data laporan | 1. Menampilkan list laporan pada datagrid |
|  | 1. Aktor memilih laporan dan menginputkan tanggal yang diinginkan. Kemudian mengklik tombol OK. | 4.Sistem menampilkan report dengan kolom dan record sesuai input dari aktor. Sistem menunggu klik aktor  untuk versi cetak.  (alternative 2.1) |
| Alternative Flow | 2.1 Kesalahan penulisan format tanggal. (kembali ke precondition) | |
| Requirement | Berhasil login sebagai pemilik | |
| Post Condition | Sistem menampilkan data laporan | |

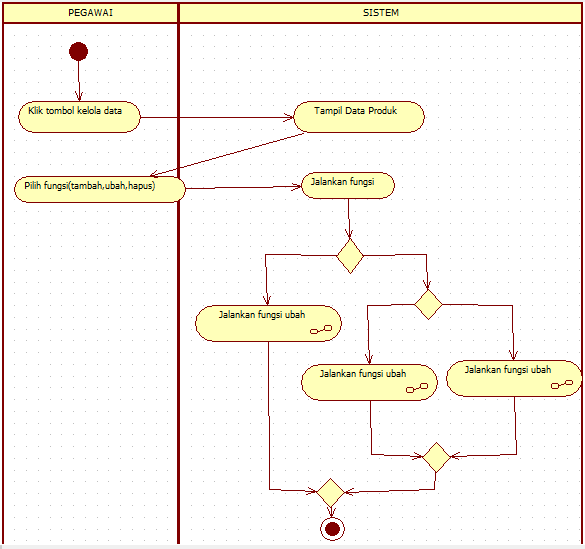
**Tabel 4.11** *Use Case Spesification* Melihat Produk

|  |  |  |
| --- | --- | --- |
| Use Case Name | Melihat Produk | |
| Aktor | Pelanggan | |
| Description | Pelanggan dapat melihat produk baik itu varian rasa,ukuran dan harga produk Makaroni Masim | |
| Precondition | *User*  telah masuk kedalam sistem dan sistem menampilkan fitur lihat produk | |
| Basic Flow | Aktor | Sistem |
| 1. *Klik*  tombol lihat produk pada bagian navbar horizontal di dalam *website* | 1. Menampilkan detail produk berupa gambar,varian rasa, ukuran serta status produk tersedia atau tidak |
| 1. Jika pelanggan berminat, *Klik* tombol pesan produk (alternative 2.1) | 1. Menampilkan form registrasi terlebih dahulu lalu ke form pemesanan produk |
| Alternative Flow | 2.1 jika pelanggan tidak berminat (kembali ke precondition) | |
| Requirement | Data produk sudah ada | |
| Post Condition | Pelangan berhasil melihat produk | |

* + 1. **Activity Diagram**

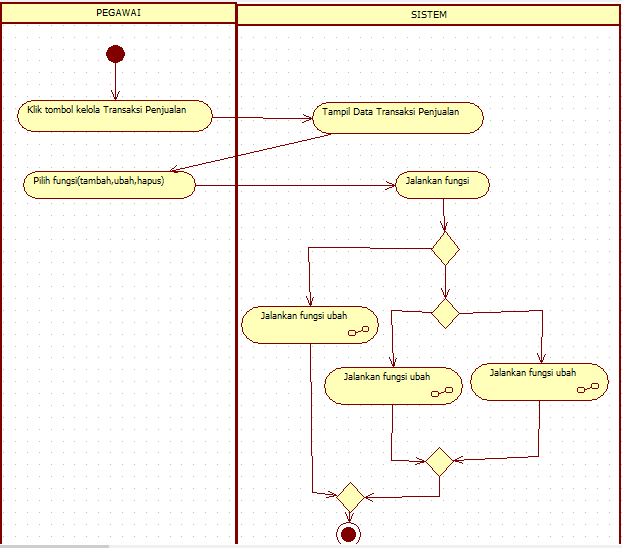
1. *Activity Diagram* Login

**Gambar 4.2** *Activity Diagram* Login

1. *Activity Diagram* Kelola Data

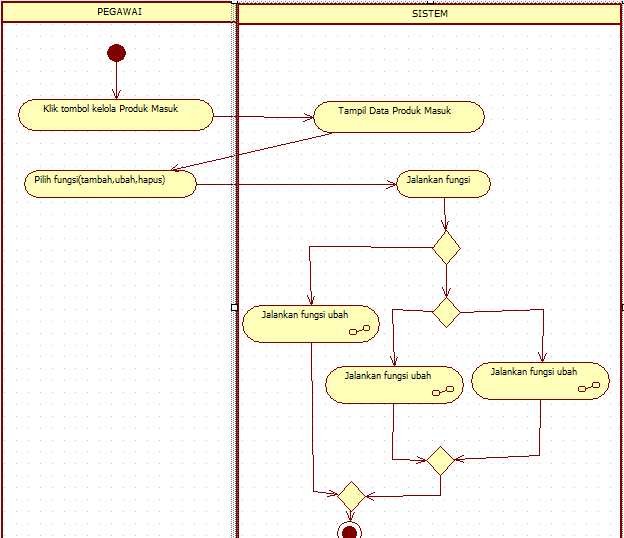
**Gambar 4.3** *Activity Diagram* Kelola Data

1. *Activity Diagram* Kelola Transaksi Penjualan

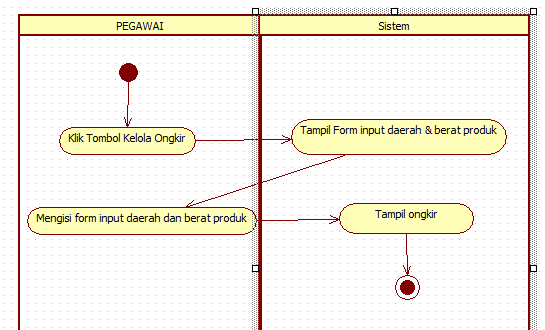


**Gambar 4.4** *Activity Diagram* Kelola Transaksi Penjualan

1. *Activity Diagram* Kelola Data Produk Masuk

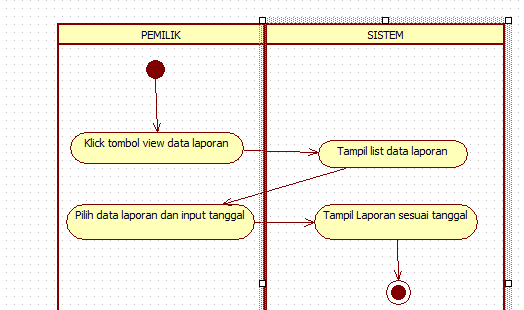


**Gambar 4.5** *Activity Diagram* Kelola Data Produk Masuk

1. *Activity Diagram* Kelola Ongkir

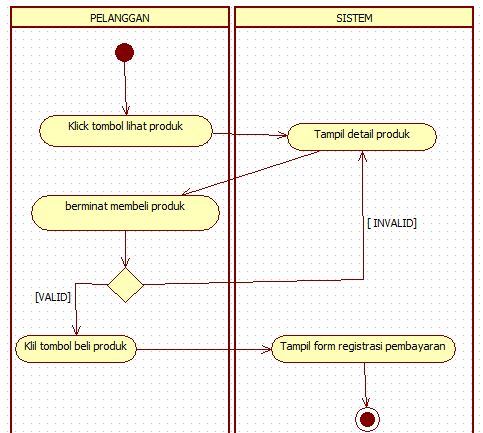
**Gambar 4.5** *Activity Diagram* Kelola Ongkir

1. *Activity Diagram* Melihat Laporan

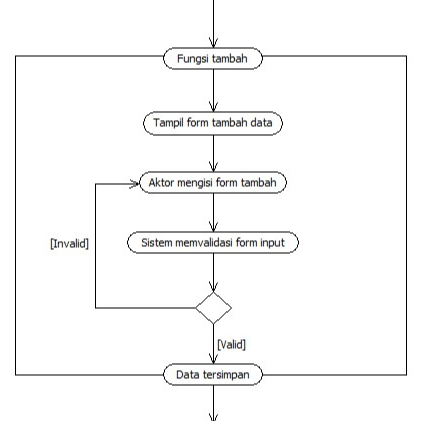


**Gambar 4.6** *Activity Diagram* Melihat Laporan

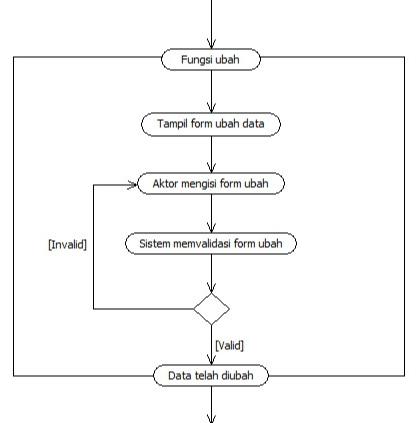
1. *Activity Diagram* Melihat Produk



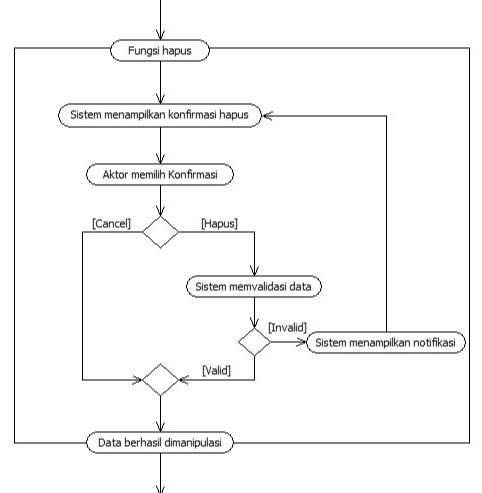
**Gambar 4.7** *Activity Diagram* Melihat Produk

* *Sub-Activity* jalankan fungsi tambah

**Gambar 4.17** *Sub-Activity* Fungsi Tambah

* *Sub-Activity* jalankan fungsi ubah

**Gambar 4.18** *Sub-Activity* Fungsi ubah

* *Sub-Activity* jalankan fungsi hapus

**Gambar 4.19** *Sub-Activity* Fungsi hapus