Software Requirements Specification

Sistem Point Of Sales

Retail Minimarket

Oleh :

Komang Arinanda

Nikolaus Aldo Halim

Blibli FUTURE Program Batch 2.0

**Agustus 2017**

**Daftar Isi**

Penjelasan sistem

Poin of Sales adalah sebuah sistem yang dapat mengelola transaksi jual beli, perhitungan akuntansi, manajemen barang dan stok, laporan laba rugi. Pada sistem ini terdapat 2 buah role user yaitu Admin dan Kasir.

Fungsionalitas yang dapat dilakukan oleh masing - masing role digambarkan dalam use case diagram berikut.

Gambar : use case diagram sistem POS minimarket

Terdapat fungsi login, jika user login sebagai Kasir maka ia dapat melakukan beberapa fungsionalitas yaitu:

1. Mengelola Akun

Fungsionalitas ini meliputi ubah foto, nama, email, dan password.

1. Mengelola Penjualan

Fungsionalitas ini meliputi input data penjualan, proses pembayaran, dan mencetak receipt.

Jika user login sebagai Admin maka ia dapat melakukan semua fungsionalitas yang dapat dilakukan oleh role Kasir. Ini dikarenakan Kasir merupakan generalisasi dari Admin. Sebagai Admin terdapat beberapa fungsional lain yang dapat dilakukan yaitu:

1. Mengelola Informasi Minimarket

Fungsionalitas ini meliputi mengubah informasi nama, alamat, nomor telp, email minimarket, dan keterangan (berisi ucapan terimakasih atau informasi promo yang ditampilkan di receipt).

1. Mengelola Pegawai

Fungsionalitas ini meliputi tambah, ubah, hapus, tampil dan cari untuk data pegawai.

1. Mengelola Supplier

Fungsionalitas ini meliputi tambah, ubah, hapus, tampil dan cari untuk data Supplier.

1. Mengelola Barang

Fungsionalitas ini meliputi tambah, ubah, hapus, tampil dan cari untuk data kategori dan data barang.

1. Mengelola Promo

Fungsionalitas ini meliputi tambah, ubah, hapus, tampil dan cari untuk data Promo atau diskon.

1. Mengelola Pembelian

Fungsionalitas ini meliputi input data pembelian produk baru.

1. Melihat Laporan

Fungsionalitas ini meliputi melihat laporan data penjualan dan pembelian dalam mingguan, bulanan, atau tahun.

Use Case Description

1. Login

|  |  |
| --- | --- |
| **Nama Use Case** | Login |
| **Aktor** | User (Admin atau Kasir) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana user melakukan login ke sistem POS |
| **Pre Condition** | User harus telah memiliki ID Pegawai dan Password yang terdata dalam sistem |
| **Flow Of Event** | 1. Use case dimulai ketika user mengakses halaman login sistem |
| 2. Sistem menampilkan halaman login dan meminta user memasukkan ID Pegawai dan Password |
| 3. User memasukkan ID Pegawai dan password |
| 4. User menekan tombol login |
| 5. Sistem melakukan pengecekan ID Pegawai dan password, |
| 6. Sistem menampilkan pesan bahwa login berhasil |
| 7. Sistem mengecek role yang dimiliki user |
| 8. Sistem menampilkan halaman awal sesuai dengan role yang dimiliki user |
| 9. Use case selesai |
| **Post Condition** | Sistem menampilkan halaman awal untuk user sesuai dengan role yang dimiliki user |
| **Alternative Flow** | Pada langkah 3, jika user memasukkan ID Pegawai atau password yang salah, sistem akan meminta user memasukkan ID Pegawai dan password yang benar |

1. Mengelola Akun

|  |  |
| --- | --- |
| **Nama Use Case** | Mengelola Akun |
| **Aktor** | User (Admin atau Kasir) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana user mengubah foto, nama, dan email. |
| **Pre Condition** | User telah diautorisasi, memiliki data baru yang akan diubah |
| **Flow Of Event** | 1. Use case dimulai ketika user memilih halaman mengelola akun 2. Sistem menampilkan halaman mengelola akun yang berisi foto, nama, dan email lama user 3. User dapat memperbarui foto atau nama dengan data baru 4. User dapat menginputkan email baru 5. Sistem melakukan validasi email 6. User menekan tombol Simpan 7. Sistem menampilkan pesan konfirmasi 8. User menekan tombol Ya pada pilihan yang muncul 9. Sistem memperbarui data user 10. Sistem menampilkan pesan berhasil 11. Use case berhasil |
| **Post Condition** | Data akun berupa foto, nama, atau email diperbarui |
| **Alternative Flow** | Pada langkah 4, jika user menginputkan email yang tidak valid maka sistem meminta inputan email yang valid  Pada langkah 8, jika user memilih tombol tidak pada pilihan yang muncul maka proses pembaruan akun dibatalkan dan proses kembali ke langkah 1 |

1. Mengubah Password

|  |  |
| --- | --- |
| **Nama Use Case** | Mengubah password |
| **Aktor** | User (Admin atau Kasir) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana user mengubah password akun |
| **Pre Condition** | User mengakses halaman mengelola akun, dan mengklik tombol ubah akun pada tampilan yang muncul, user memiliki data password lama dan password baru |
| **Flow Of Event** | 1. Use case dimulai ketika user mengakses halaman mengelola akun dan mengklik tombol ubah password 2. Sistem menampilkan form ubah password 3. User menginputkan password lama, password baru, dan input ulang password baru 4. User mengklik tombol simpan 5. Sistem memvalidasi password lama 6. Sistem memvalidasi password baru dengan inputan ulang password 7. Sistem menampilkan pesan konvirmasi ubah password 8. User mengklik tombol Ya 9. Sistem memperbarui password 10. Sistem menampilkan pesan berhasil 11. Use case selesai |
| **Post Condition** | Password pada sistem diperbaharui |
| **Alternative Flow** | Pada langkah 3, jika user menginputkan password lama yang salah maka sistem meminta inputan password lama yang benar  Pada langkah 3, jika user menginputkan password baru dan inputan ulang password baru yang tidak sama persis maka sistem meminta password baru yang benar  Pada langkah 8 jika user mengklik tombol Tidak pada pilihan yang muncul maka proses perbarui password dibatalkan dan proses kembali ke langkah 1 |

2. Mengelola Informasi Minimarket

|  |  |
| --- | --- |
| **Nama Use Case** | Mengelola Informasi Minimarket |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin mengelola informasi nama, alamat, nomor telp, email, dan keterangan (receipt) minimarket |
| **Pre Condition** | Admin harus sudah diotorisasi oleh sistem |
| **Flow Of Event** | 1. Use case dimulai ketika Admin masuk ke halaman Pengelolaan Informasi Minimarket  2. Sistem menampilkan informasi logo, nama, alamat, nomot telp, email dan keterangan (receipt) lama dan meminta inputan informasi logo, nama, alamat, nomot telp, email dan keterangan (receipt) yang baru  3. Admin menginputkan logo, nama, alamat, nomot telp, email dan keterangan (receipt) yang baru kemudian mengklik tombol Simpan  4. Sistem menampilkan pesan konfirmasi update  5. Admin melakukan konfirmasi update dengan memilik Ya pada sistem  6. Sistem memperbarui informasi nama, alamat, nomot telp, email dan keterangan (receipt)  7. Use case selesai |
| **Post Condition** | Sistem memperbarui informasi nama, alamat, nomot telp, email dan keterangan (receipt) |
| **Alternative Flow** | Pada langkah 5, jika Admin memilih Tidak mengkonfirmasi update maka proses update dibatalkan dan use case selesai |

3. Mengelola Barang

3.1 Mengelola Kategori

a. Tambah kategori

|  |  |
| --- | --- |
| **Nama Use Case** | Tambah Kategori |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan sebuah kategori baru |
| **Pre Condition** | Admin memilih tampilan tambah kategori dan memiliki data kategori baru yang akan ditambahkan |
| **Flow Of Event** | 1. Use case dimulai ketika Admin memilih tampilan tambah kategori  2. Sistem meminta inputan Kode Kategori baru  3. Admin menginputkan Kode Kategori Baru  4. Sistem melakukan pengecekan keunikan kode kategori baru  5. Admin menginputkan nama kategori baru  6. Admin menginputkan deskripsi kategori  7. Sistem menampilkan pesan konfirmasi tambah kategori  8. User memilih Ya pada pilihan yang muncul  9. Sistem mencatat data kategori baru  10. Use case selesai |
| **Post Condition** | Sistem mencatat data kategori baru |
| **Alternative Flow** | Pada langkah 3, jika Admin menginputkan kode kategori yang tidak unik maka sistem menampilkan pesan kesalahan dan kemudian meminta Admin menginputkan kode kategori yang baru dan unik  Pada langkah nomor 7, jika Admin memilih “Tidak” pada pilihan yang muncul maka sistem membatalkan penambahan kategori dan proses kembali ke langkah 1 |

b. Cari Kategori

|  |  |
| --- | --- |
| **Nama Use Case** | Cari Kategori |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melakukan proses pencarian kategori |
| **Pre Condition** | Admin memilih tampilan mengelola kategori dan memiliki kode atau nama kategori yang akan dicari |
| **Flow Of Event** | 1. Use case dimulai ketika Admin akan melakukan pengubahan pada data kategori ataupun menghapus kategori tertentu  2. Sistem meminta inputan berupa Kode atau Nama kategori yang dicari  3. Admin menginputkan Kode atau Nama Kategori yang dicari  4. Sistem melakukan pencarian kategori  5. Sistem menampilkan Kode, Nama, dan deskripsi kategori  6. Use case selesai |
| **Post Condition** | Sistem menampilkan kode, nama, dan deskripsi kategori |
| **Alternative Flow** | Pada langkah 4, jika sistem tidak menemukan kode atau nama kategori yang dicari maka sistem akan menampilkan pesan kesalahan bahwa kode atau nama kategori yang dicari tidak ditemukan |

c. Mengubah Kategori

|  |  |
| --- | --- |
| **Nama Use Case** | Mengubah Kategori |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin mengubah data kategori tertentu |
| **Pre Condition** | Admin memilih tampilan mengubah kategori dan memiliki kode atau nama kategori yang akan diubah serta nama atau deskripsi baru |
| **Flow Of Event** | 1. Use case dimulai ketika Admin menginputkan kode atau nama kategori yang akan diubah datanya  2. Sistem melakukan pencarian data kategori  3. Admin menginputkan nama kategori yang baru  4. Admin menginputkan deskripsi kategori yang baru  5. Sistem menampilkan pesan konfirmasi update  6. Admin memilih “Ya” pada pilihan yang ada  7. Sistem memperbarui nama dan deskripsi kategori  8. Use case selesai |
| **Post Condition** | Sistem memperbarui nama dan deskripsi kategori |
| **Alternative Flow** | Pada langkah 6, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses mengubah kategori dibatalkan dan proses kembali ke langkah 1 |

d. Menghapus Kategori

|  |  |
| --- | --- |
| **Nama Use Case** | Menghapus Kategori |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menghapus data kategori tertentu |
| **Pre Condition** | Admin memilih tampilan menghapus kategori dan memiliki kode atau nama kategori yang akan dihapus |
| **Flow Of Event** | 1. Use case dimulai ketika Admin menginputkan kode atau nama kategori yang akan dihapus  2. Sistem melakukan pencarian data kategori  4. Sistem menampilkan pesan konfirmasi hapus  5. Admin memilih “Ya” pada pilihan yang ada  6. Sistem menghapus data kategori  7. Use case selesai |
| **Post Condition** | Sistem menghapus data kategori |
| **Alternative Flow** | Pada langkah 5, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses menghapus kategori dibatalkan dan proses kembali ke langkah 1 |

3.2. Mengelola Barang

a. Tambah Barang

|  |  |
| --- | --- |
| **Nama Use Case** | Tambah Barang |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan sebuah barang baru kedalam sistem POS |
| **Pre Condition** | Admin memilih tampilan tambah barang dan memiliki data barang baru yang akan ditambahkan |
| **Flow Of Event** | 1. Use case dimulai ketika sistem meminta input kode barang baru  2. Sistem mengecek keunikan kode barang  3. Admin menginputkan nama kategori barang  4. Admin menginputkan nama barang  5. Admin menginputkan deskripsi barang  6. Sistem menginisialisasi stok dengan nilai 0  7. Sistem menginisialisasi harga dengan nilai 0  8. Sistem menampilkan pesan konfirmasi tambah barang  9. User memilih Ya pada pilihan yang muncul  10. Sistem mencatat data barang baru  11. Use case selesai |
| **Post Condition** | Sistem mencatat data barang baru |
| **Alternative Flow** | Pada langkah 1, jika Admin menginputkan kode barang yang tidak unik atau sudah terdata dalam sistem maka sistem menampilkan pesan kesalahan dan kemudian meminta Admin menginputkan kode barang yang baru dan unik  Pada langkah nomor 9, jika Admin memilih “Tidak” pada pilihan yang muncul maka sistem membatalkan penambahan barang dan proses kembali ke langkah 1 |

b. Cari Barang

|  |  |
| --- | --- |
| **Nama Use Case** | Cari Barang |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melakukan proses pencarian barang |
| **Pre Condition** | Admin memilih tampilan mengelola barang dan memiliki kode atau nama barang yang akan dicari |
| **Flow Of Event** | 1. Use case dimulai ketika Admin akan melakukan pengubahan pada data barang ataupun menghapus barang tertentu  2. Sistem meminta inputan berupa Kode atau Nama kategori yang dicari  3. Admin menginputkan Kode atau Nama Barang yang dicari  4. Sistem melakukan pencarian barang  5. Sistem menampilkan Kode, Nama Kategori, Nama Barang, deskripsi barang, stok, dan harga barang  6. Use case selesai |
| **Post Condition** | Sistem menampilkan kode, nama, dan deskripsi kategori |
| **Alternative Flow** | Pada langkah 4, jika sistem tidak menemukan kode atau nama barang yang dicari maka sistem akan menampilkan pesan kesalahan bahwa kode atau nama barang yang dicari tidak ditemukan |

c. Mengubah Barang

|  |  |
| --- | --- |
| **Nama Use Case** | Mengubah Barang |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin mengubah data barang tertentu |
| **Pre Condition** | Admin memilih tampilan mengubah barang dan memiliki kode atau nama barang yang akan diubah serta nama kategori, nama barang, deskripsi atau harga baru |
| **Flow Of Event** | 1. Use case dimulai ketika Admin menginputkan kode atau nama barang yang akan diubah  2. Sistem melakukan pencarian data barang  3. Admin menginputkan nama kategori yang baru  4. Admin menginputkan nama barang yang baru  5. Admin menginputkan deskripsi barang yang baru  4. Admin menginputkan harga barang yang baru  5. Sistem menampilkan pesan konfirmasi update  6. Admin memilih “Ya” pada pilihan yang ada  7. Sistem memperbarui nama kategori, nama barang, deskripsi barang atau harga barang  8. Use case selesai |
| **Post Condition** | Sistem memperbarui nama kategori, nama barang, deskripsi barang atau harga barang |
| **Alternative Flow** | Pada langkah 6, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses mengubah barang dibatalkan dan proses kembali ke langkah 1 |

d. Menghapus Barang

|  |  |
| --- | --- |
| **Nama Use Case** | Menghapus Barang |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menghapus data barang tertentu |
| **Pre Condition** | Admin memilih tampilan menghapus barang dan memiliki kode atau nama barang yang akan dihapus |
| **Flow Of Event** | 1. Use case dimulai ketika Admin menginputkan kode atau nama barang yang akan dihapus  2. Sistem melakukan pencarian data barang  4. Sistem menampilkan pesan konfirmasi hapus  5. Admin memilih “Ya” pada pilihan yang ada  6. Sistem menghapus data barang  7. Use case selesai |
| **Post Condition** | Sistem menghapus data barang |
| **Alternative Flow** | Pada langkah 5, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses menghapus barang dibatalkan dan proses kembali ke langkah 1 |

4.