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**

Daftar Isi........................................................................................................................................2

Penjelasan Sistem..........................................................................................................................3

Use Case Description.....................................................................................................................5

A. Mengelola Akun ...........................................................................................................5

1. Login................................................................................................................5

2. Mengelola Akun...............................................................................................6

3. Mengubah Password.......................................................................................7

B. Mengelola Informasi Minimarket.................................................................................8

C. Mengelola Pegawai.......................................................................................................9

1. Tampil Pegawai................................................................................................9

2. Tambah Pegawai............................................................................................10

3. Mengubah Pegawai.......................................................................................11

4. Menghapus Pegawai......................................................................................12

D. Mengelola Produk......................................................................................................13

1. Mengelola Kategori........................................................................................13

a. Tampil Kategori.................................................................................13

b. Tambah Kategori...............................................................................14

c. Mengubah Kategori...........................................................................15

d. Menghapus Kategori.........................................................................16

2. Mengelola Produk..........................................................................................17

a. Tampil Produk...................................................................................17

b. Tambah Produk.................................................................................18

c. Mengubah Produk.............................................................................19

d. Menghapus Produk...........................................................................20

E. Mengelola Stok...........................................................................................................21

1. Tampil Stok Produk........................................................................................21

2. Menambah Stok.............................................................................................22

F. Melakukan Transaksi...................................................................................................23

1. Input Data Transaksi.......................................................................................23

2. Hapus Record Transaksi.................................................................................24

3. Batal Transaksi...............................................................................................25

4. Bayar..............................................................................................................26

G. Mengelola Promo.......................................................................................................27

1. Tampil Promo.................................................................................................27

2. Tambah Promo...............................................................................................28

a. Tambah Promo Diskon Produk..........................................................28

b. Tambah Promo Diskon Total Belanja.................................................29

c. Tambah Promo Buy x Get Y................................................................30

3. Mengubah Promo..........................................................................................31

4. Menghapus Promo.........................................................................................32

H. Melihat Laporan.........................................................................................................33

1. Laporan Transaksi..........................................................................................33

2. Laporan Statistik Penjualan............................................................................34

Penjelasan Sistem

Poin of Sales adalah sebuah sistem yang dapat mengelola transaksi jual beli, perhitungan akuntansi, manajemen produk 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. Melakukan Transaksi

Fungsionalitas ini meliputi input data transaksi, 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 Produk

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

1. Mengelola Stok

Fungsionalitas ini digunakan untuk penambahan stok

1. Mengelola Promo

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

1. Melihat Laporan

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

Use Case Description

**A. Mengelola Akun**

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 |

**B. 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, pajak, 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, pajak 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 |

**C. Mengelola Pegawai**

1. Tampil Pegawai

|  |  |
| --- | --- |
| **Nama Use Case** | Tampil Pegawai |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melakukan proses mencari dan menampilkan pegawai |
| **Pre Condition** | Admin memilih halaman mengelola pegawai, memiliki ID atau nama pegawai yang akan dicari dan tampilkan |
| **Flow Of Event** | 1. Use case dimulai ketika Admin akan melihat, mengubah, ataupun menghapus pegawai tertentu dan mengakses halaman mengelola pegawai  2. Sistem meminta inputan berupa ID atau Nama pegawai yang dicari  3. Admin menginputkan ID atau Nama pegawai yang dicari  4. Sistem melakukan pencarian pegawai  5. Sistem menampilkan ID, Nama, Role, dan email pegawai  6. Use case selesai |
| **Post Condition** | Sistem menampilkan ID, Nama, Role, dan email pegawai |
| **Alternative Flow** | Admin dapat melakukan Export hasil kedalam bentuk file dokumen dengan mengklik button Export, Sistem akan mengeksport data kategori kedalam file dokumen.  Pada langkah 4, jika sistem tidak menemukan ID atau nama pegawai yang dicari maka sistem akan menampilkan pesan kesalahan bahwa ID atau nama pegawai yang dicari tidak ditemukan |

2. Tambah Pegawai

|  |  |
| --- | --- |
| **Nama Use Case** | Tambah Pegawai |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan data pegawai baru |
| **Pre Condition** | Admin memilih halaman mengelola pegawai dan memiliki data pegawai baru yang akan ditambahkan |
| **Flow Of Event** | 1. Use case dimulai ketika Admin memilih button tambah pada halaman mengelola pegawai  2. Sistem meminta inputan ID pegawai baru  3. Sistem melakukan pengecekan keunikan kode pegawai baru  4. Admin menginputkan nama pegawai  5. Admin menginputkan role pegawai  6. Sistem menset password dengan “password”  7. Admin menginputkan email pegawai  8. Admin mengklik button Tambah  9. Sistem menampilkan pesan konfirmasi tambah pegawai  10. Admin memilih Ya pada pilihan yang muncul  11. Sistem mencatat data pegawai baru  12. Use case selesai |
| **Post Condition** | Sistem mencatat data kategori baru |
| **Alternative Flow** | Pada langkah 3, jika Admin menginputkan ID pegawai yang tidak unik maka sistem menampilkan pesan kesalahan dan kemudian meminta Admin menginputkan ID yang baru dan unik  Pada langkah nomor 9, jika Admin memilih “Tidak” pada pilihan yang muncul maka sistem membatalkan penambahan pegawai dan proses kembali ke langkah 1 |

3. Mengubah Pegawai

|  |  |
| --- | --- |
| **Nama Use Case** | Mengubah Pegawai |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin mengubah data pegawai tertentu |
| **Pre Condition** | Admin memilih halaman mengelola pegawai dan memiliki ID atau nama pegawai yang akan diubah serta data yang akan diubah |
| **Flow Of Event** | * + - 1. Use case dimulai ketika Admin memilih pegawai yang akan diubah (dapat melakukan pencarian terlebih dahulu)  1. Sistem melakukan pencarian data pegawai 2. Admin mengklik button Ubah 3. Sistem menampilkan form ubah pegawai 4. Admin menginputkan nama baru 5. Admin menginputkan role baru 6. Admin menginputkan email baru 7. Sistem menampilkan pesan konfirmasi update 8. Admin memilih “Ya” pada pilihan yang ada 9. Sistem memperbarui nama, role, dan email 10. Use case selesai |
| **Post Condition** | Sistem memperbarui nama, role, dan email |
| **Alternative Flow** | Pada langkah 9, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses mengubah data pegawai dibatalkan dan proses kembali ke langkah 1 |

4. Menghapus Pegawai

|  |  |
| --- | --- |
| **Nama Use Case** | Menghapus Pegawai |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menghapus data pegawai tertentu |
| **Pre Condition** | Admin memilih halaman mengelola pegawai dan memiliki ID atau nama pegawai yang akan dihapus |
| **Flow Of Event** | 1. Use case dimulai ketika Admin mengklik button hapus pada halaman mengelola pegawai  2. Admin memilih data pegawai yang akan dihapus (dapat dilakukan dengan pencarian)  3. Sistem melakukan pencarian data pegawai  4. Sistem menampilkan pesan konfirmasi hapus  5. Admin memilih “Ya” pada pilihan yang ada  6. Sistem menghapus data pegawai  7. Use case selesai |
| **Post Condition** | Sistem menghapus data pegawai |
| **Alternative Flow** | Pada langkah 5, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses menghapus kategori dibatalkan dan proses kembali ke langkah 1 |

**D. Mengelola Produk**

1. Mengelola Kategori

a. Tampil Kategori

|  |  |
| --- | --- |
| **Nama Use Case** | Tampil Kategori |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melakukan proses mencari dan menampilkan kategori |
| **Pre Condition** | Admin memilih halaman mengelola kategori, memiliki kode atau nama kategori yang akan dicari dan tampilkan |
| **Flow Of Event** | 1. Use case dimulai ketika Admin akan melihat, 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** | Admin dapat melakukan Export tampilan kedalam bentuk file dokumen dengan mengklik button Export, Sistem akan mengeksport data kategori kedalam file dokumen.  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 |

b. Tambah kategori

|  |  |
| --- | --- |
| **Nama Use Case** | Tambah Kategori |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan sebuah kategori baru |
| **Pre Condition** | Admin memilih halaman mengelola 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 8, jika Admin memilih “Tidak” pada pilihan yang muncul maka sistem membatalkan penambahan kategori dan proses kembali ke langkah 1 |

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 halaman mengelola kategori dan memiliki kode atau nama kategori yang akan diubah serta nama atau deskripsi baru |
| **Flow Of Event** | * + - 1. Use case dimulai ketika Admin memilih kotegori yang akan diubah, juga dapat melakukan pencarian terlebih dahulu  1. Sistem melakukan pencarian data kategori 2. Admin mengklik button Ubah 3. Sistem menampilkan form ubah kategori 4. Admin menginputkan nama kategori yang baru 5. Admin menginputkan deskripsi kategori yang baru 6. Sistem menampilkan pesan konfirmasi update 7. Admin memilih “Ya” pada pilihan yang ada 8. Sistem memperbarui nama dan deskripsi kategori 9. 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 halaman mengelola kategori dan memiliki kode atau nama kategori yang akan dihapus, tidak ada data produk pada 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  3. Sistem memeriksa keberadaan data produk pada kategori yang akan dihapus  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 3, jika ada data produk dalam kategori tersebut maka proses hapus kategori gagal dan proses kembali ke langkah 1  Pada langkah 5, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses menghapus kategori dibatalkan dan proses kembali ke langkah 1 |

2. Mengelola Produk

a. Tampil Produk

|  |  |
| --- | --- |
| **Nama Use Case** | Tampil Produk |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melakukan proses pencarian dan menampilkan Produk |
| **Pre Condition** | Admin memilih halaman mengelola produk dan memiliki kode atau nama produk yang akan dicari dan tampilkan |
| **Flow Of Event** | 1. Use case dimulai ketika Admin akan melihat, mengubah, atau menghapus produk tertentu  2. Sistem meminta inputan berupa Kode atau Nama kategori yang dicari  3. Admin menginputkan Kode atau Nama produk yang dicari  4. Sistem melakukan pencarian produk  5. Sistem menampilkan Kode, Nama Kategori, Nama produk, deskripsi produk, stok, dan harga produk  6. Use case selesai |
| **Post Condition** | Sistem menampilkan kode, nama kategori, nama produk,deskripsi, stok, dan harga produk |
| **Alternative Flow** | Admin dapat melakukan Export hasil kedalam bentuk file dokumen dengan mengklik button Export. Sistem akan mengeksport data kategori kedalam file dokumen.  Pada langkah 4, jika sistem tidak menemukan kode atau nama produk yang dicari maka sistem akan menampilkan pesan kesalahan bahwa kode atau nama produk yang dicari tidak ditemukan |

b. Tambah Produk

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

c. Mengubah Produk

|  |  |
| --- | --- |
| **Nama Use Case** | Mengubah Produk |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin mengubah data produk tertentu |
| **Pre Condition** | Admin memilih halaman mengelola produk dan memiliki kode atau nama produk yang akan diubah serta nama kategori, nama produk, atau deskripsi baru |
| **Flow Of Event** | 1. Use case dimulai ketika Admin memilih produk yang akan diubah yang dapat dilakukan dengan melakukan pencarian terlebih dahulu  2. Sistem melakukan pencarian data produk  3. Admin mengklik button Ubah  4. Sistem menampilkan form ubah produk  4. Admin menginputkan nama kategori yang baru  5. Admin menginputkan nama produk yang baru  6. Admin menginputkan deskripsi produk yang baru  8. Sistem menampilkan pesan konfirmasi update  9. Admin memilih “Ya” pada pilihan yang ada  10. Sistem memperbarui nama kategori, nama produk, deskripsi produk .  11. Use case selesai |
| **Post Condition** | Sistem memperbarui nama kategori, nama produk dan deskripsi produk |
| **Alternative Flow** | Pada langkah 6, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses mengubah produk dibatalkan dan proses kembali ke langkah 1 |

d. Menghapus Produk

|  |  |
| --- | --- |
| **Nama Use Case** | Menghapus Produk |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menghapus data produk tertentu |
| **Pre Condition** | Admin memilih halaman mengelola produk dan memiliki kode atau nama produk yang akan dihapus |
| **Flow Of Event** | 1. Use case dimulai ketika Admin menginputkan kode atau nama produk yang akan dihapus  2. Sistem melakukan pencarian data produk  3. Sistem memastikan bahwa stok produk kosong  4. Sistem menampilkan pesan konfirmasi hapus  5. Admin memilih “Ya” pada pilihan yang ada  6. Sistem menghapus data produk  7. Use case selesai |
| **Post Condition** | Sistem menghapus data produk |
| **Alternative Flow** | Pada langkah 3, jika produk yang akan dihapus memiliki stok maka proses hapus kategori gagal dan proses kembali ke langkah 1  Pada langkah 5, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses menghapus produk dibatalkan dan proses kembali ke langkah 1 |

**E. Mengelola Stok**

1. Tampil Stok Produk

|  |  |
| --- | --- |
| **Nama Use Case** | Tampil Stok Produk |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melakukan proses pencarian dan menampilkan stok Produk |
| **Pre Condition** | Admin memilih halaman mengelola stok dan memiliki kode atau nama produk yang akan dicari dan tampilkan |
| **Flow Of Event** | * + - 1. Use case dimulai ketika Admin menginputkan kode, nama, atau kategori produk yang akan ditampilkan data stoknya  1. Sistem melakukan pencarian produk 2. Sistem menampilkan kode, nama kategori, nama produk, dan stok 3. Use case selesai |
| **Post Condition** | Sistem menampilkan kode, nama kategori, nama produk, dan stok |
| **Alternative Flow** | Admin dapat melakukan Export hasil kedalam bentuk file dokumen dengan mengklik button Export. Sistem akan mengeksport data stok kedalam file dokumen.  Pada langkah 2, jika sistem tidak menemukan kode, nama, atau produk yang dicari maka sistem akan menampilkan pesan kesalahan bahwa kode atau nama produk yang dicari tidak ditemukan |

2. Menambah Stok

|  |  |
| --- | --- |
| **Nama Use Case** | Menambah Stok |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan stok untuk produk tertentu |
| **Pre Condition** | Admin memilih halaman mengelola stok dan memiliki kode atau nama produk yang akan ditambahkan stoknya |
| **Flow Of Event** | * + - 1. Use case dimulai ketika Admin menginputkan kode atau nama produk yang akan ditambahkan stoknya  1. Sistem melakukan pencarian data produk 2. Admin menginputkan jumlah produk 3. Sistem melakukan validasi jumlah dan menampilkan pesan konfirmasi 4. Admin memilih “Ya” pada pilihan yang muncul 5. Sistem menambahkan jumlah stok produk 6. Use case selesai |
| **Post Condition** | Sistem menambahkan stok produk |
| **Alternative Flow** | Pada langkah 1, jika inputan kode produk tidak terdata dalam sistem maka proses tambah stok tidak dapat dilakukan dan meminta kode produk yang telah terdata.  Pada langkah 3, jika inputan jumlah tidak valid maka sistem meminta inputan ulang yang valid  Pada langkah 5, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses menghapus produk dibatalkan dan proses kembali ke langkah 1 |

**F. Melakukan Transaksi**

1. Input Data Transaksi

|  |  |
| --- | --- |
| **Nama Use Case** | Input Data Transaksi |
| **Aktor** | User (Kasir/Admin) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana User menambahkan data transaksi |
| **Pre Condition** | Admin memilih halaman Transaksi dan memiliki kode atau nama produk yang akan ditambahkan ke data transaksi |
| **Flow Of Event** | 1. Use case dimulai ketika User memilih halaman Transaksi 2. Sistem men-generate Nomor Transaksi 3. Sistem mengeset waktu/Tanggal, Nama dan ide user yang sedang melakukan proses transaksi 4. User menginputkan kode produk 5. Sistem melakukan pencarian data produk dan menampilkan nama produk 6. User menginputkan jumlah produk 7. Sistem memvalidasi jumlah inputan dengan ketersediaan stok dan menampilkan pesan konfirmasi tambah data transaksi 8. User mengklik button tambah atau menekan + pada keyboard 9. Sistem mengambil harga produk, menghitung diskon, menghitung sub total 10. Sistem menambahkan data produk 11. Sistem menampilkan data pada tabel, menghitung pajak, dan total akhir 12. Sistem menampilkan Total Akhir dan pesan tambah produk atau tidak 13. User memilih tidak 14. Use case selesai |
| **Post Condition** | Sistem menambahkan stok produk |
| **Alternative Flow** | Pada langkah 3, jika waktu/tanggal tidak sesuai maka user dapat mengubah dengan waktu yang sesuai.  Pada langkah 4, jika sistem tidak menemukan data yang diinputkan maka sistem meminta inputan ulang  Pada langkah 7, jika inputan jumlah tidak valid maka sistem meminta inputan ulang yang valid  Pada langkah 13, jika User memilih “Ya” pada pilihan yang muncul maka proses berlanjut kembali ke langkah 4 |

2. Hapus Record Transaksi

|  |  |
| --- | --- |
| **Nama Use Case** | Menghapus Record Transaksi |
| **Aktor** | User(Kasir/Admin) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana User menghapus record transaksi tertentu |
| **Pre Condition** | User telah menginputkan data transaksi sebelumnya dan mengetahui kode/ nama produk yang akan dihapus dari daftar transaksi |
| **Flow Of Event** | * + - 1. Use case dimulai ketika memilih data dengan mengklik pada record data yang akan dihapus kemudian mengklik button hapus atau menekan – pada keyboard       2. Sistem mendapatkan kode produk kemudian melakukan pencarian record produk tersebut  1. Sistem menampilkan pesan konfirmasi hapus 2. User memilih “Ya” pada pilihan yang muncul 3. Sistem menghapus record transaksi 4. Use case selesai |
| **Post Condition** | Sistem menghapus record transaksi |
| **Alternative Flow** | Pada langkah 4, jika User memilih “Tidak” pada pilihan yang muncul maka proses menghapus record transaksi dibatalkan dan proses kembali ke langkah 1 |

3. Batal Transaksi

|  |  |
| --- | --- |
| **Nama Use Case** | Batal Transaksi |
| **Aktor** | User(Kasir/Admin) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana User membatalkan transaksi yang sedang berjalan |
| **Pre Condition** | User telah menginputkan data transaksi sebelumnya |
| **Flow Of Event** | 1. Use case dimulai ketika User mengklik button Batal atau menekan ESC pada keyboard 2. Sistem menampilkan pesan konfirmasi pembatalan transaksi 3. User memilih “Ya” pada pilihan yang muncul 4. Sistem menghapus semua record transaksi, mengosongkan tabel, mengosongkan field kode, nama, dan jumlah produk 5. Use case selesai |
| **Post Condition** | Sistem menghapus semua record transaksi, mengosongkan tabel, mengosongkan field kode, nama, dan jumlah produk |
| **Alternative Flow** | Pada langkah 3, jika User memilih “Tidak” pada pilihan yang muncul maka proses menghapus record transaksi dibatalkan dan proses kembali ke langkah 1 |

4. Bayar

|  |  |
| --- | --- |
| **Nama Use Case** | Bayar |
| **Aktor** | User(Kasir/Admin) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana User melakukan proses pembayaran |
| **Pre Condition** | User telah menginputkan data-data transaksi |
| **Flow Of Event** | 1. Use case dimulai ketika User mengklik button Bayar atau menekan \* pada keyboard 2. Sistem meminta inputan jenis metode pembayaran(Tunai/ Debit/Kredit) 3. User menginputkan jenis metode Tunai 4. User menginputkan nominal bayar 5. Sistem memastikan nominal bayar tidak lebih kecil dari total akhir 6. Sistem menghitung dan menampilkan nominal kembalian 7. User mengklik button Bayar atau menekan tombol \* pada keyboard 8. Sistem menampilkan pesan konfirmasi Bayar 9. User memilih “Ya” pada pilihan yang muncul 10. Sistem menyimpan data transaksi dan mengurangi stok-stok produk terjual 11. Sistem mengenerate dan mencetak receipt 12. Use case selesai |
| **Post Condition** | Sistem menyimpan data transaksi dan mengurangi stok-stok produk |
| **Alternative Flow** | Pada langkah 3, jika user memilih jenis metode Debit atau Kredit maka Sistem meminta inputan nomor kartu dan proses langsung ke langkah 7  Pada langkah 9, jika User memilih “Tidak” pada pilihan yang muncul maka proses menghapus record transaksi dibatalkan dan proses kembali ke langkah 1 |

**G. Mengelola Promo**

1. Tampil Promo

|  |  |
| --- | --- |
| **Nama Use Case** | Tampil Promo |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melakukan proses pencarian dan menampilkan promo ataupun export hasil |
| **Pre Condition** | Admin memilih halaman promo dan memiliki kode promo yang akan dicari dan tampilkan |
| **Flow Of Event** | 1. Use case dimulai ketika Admin akan melihat, mengubah, atau menghapus promo tertentu  2. Sistem meminta inputan berupa Kode dan jenis promo  3. Admin menginputkan Kode promo  4. Admin memilih jenis promo Diskon Produk  5. Sistem melakukan pencarian pencarian  6. Sistem menampilkan Kode, nama produk , tanggal mulai, berakhir, diskon, dan status  7. Use case selesai |
| **Post Condition** | Sistem menampilkan Kode, nama produk , tanggal mulai, berakhir, diskon, dan status |
| **Alternative Flow** | Admin dapat melakukan Export hasil kedalam bentuk file dokumen dengan mengklik buttn Export. Sistem akan mengeksport data kategori kedalam file dokumen.  Pada langkah 5, jika sistem tidak menemukan kode atau nama produk yang dicari maka sistem akan menampilkan pesan kesalahan bahwa kode atau nama produk yang dicari tidak ditemukan  Pada langkah 4 jika admin memilih jenis Diskon Total Belanja maka data yang ditampilkan adalah kode, tanggal mulai, berakhir, belanja minimal, diskon, dan status.  Pada langkah 4 jika admin memilih jenis Diskon Buy x get y maka data yang ditampilkan adalah kode, tanggal mulai, berakhir, nama produk x, jumlah min, nama produk y, jumlah, dan status |

2. Tambah Promo

a. Tambah promo diskon produk

|  |  |
| --- | --- |
| **Nama Use Case** | Tambah Promo diskon produk |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan promo diskon produk |
| **Pre Condition** | Admin memilih halaman promo dan mengklik button Tambah, Admin memiliki data diskon yang akan ditambahkan |
| **Flow Of Event** | 1. Use case dimulai ketika sistem menampilkan form tambah diskon 2. Admin menginputkan Kode Promo 3. Sistem memvalidasi keunikan kode 4. Admin menginputkan tanggal mulainya promo 5. Admin menginputkan tanggal berakhir promo 6. Sistem memvalidasi tanggal mulai dan berakhir 7. Admin memilih jenis promo Diskon Produk 8. Admin menginput kode produk promo 9. Sistem menampilkan nama produk promo 10. Admin menginput persentase diskon 11. Sistem memvalidasi persentase diskon 12. Sistem menampilkan pesan konfirmasi tambah diskon 13. Admin memilih Ya pada pilihan yang muncul 14. Sistem mencatat data diskon 15. Use case selesai |
| **Post Condition** | Sistem mencatat data promo Diskon Produk |
| **Alternative Flow** | Pada langkah 3, 6, dan 11 jika sistem menilai bahwa inputan tidak valid mana sistem menampilkan pesan kesalahan dan meminta input ulang  Pada langkah nomor 13, jika Admin memilih “Tidak” pada pilihan yang muncul maka sistem membatalkan penambahan data promo dan proses kembali ke langkah 1 |

b. Tambah Promo Diskon Total Belanja

|  |  |
| --- | --- |
| **Nama Use Case** | Tambah Promo diskon total belanja |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan promo total belanja |
| **Pre Condition** | Admin memilih halaman promo dan mengklik button Tambah, Admin memiliki data diskon yang akan ditambahkan |
| **Flow Of Event** | 1. Use case dimulai ketika sistem menampilkan form tambah diskon 2. Admin menginputkan Kode Promo 3. Sistem memvalidasi keunikan kode 4. Admin menginputkan tanggal mulainya promo 5. Admin menginputkan tanggal berakhir promo 6. Sistem memvalidasi tanggal mulai dan berakhir 7. Admin memilih jenis promo Diskon Total Belanja 8. Admin menginput total belanja minimal 9. Admin menginput persentase diskon 10. Sistem memvalidasi persentase diskon 11. Sistem menampilkan pesan konfirmasi tambah diskon 12. Admin memilih Ya pada pilihan yang muncul 13. Sistem mencatat data diskon 14. Use case selesai |
| **Post Condition** | Sistem mencatat data promo Diskon Produk |
| **Alternative Flow** | Pada langkah 3, 6, dan 10 jika sistem menilai bahwa inputan tidak valid mana sistem menampilkan pesan kesalahan dan meminta input ulang  Pada langkah nomor 13, jika Admin memilih “Tidak” pada pilihan yang muncul maka sistem membatalkan penambahan data promo dan proses kembali ke langkah 1 |

c. Tambah Promo Buy x Get y

|  |  |
| --- | --- |
| **Nama Use Case** | Tambah Promo Buy x Get y |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menambahkan promo Buy x Get y |
| **Pre Condition** | Admin memilih halaman promo dan mengklik button Tambah, Admin memiliki data diskon yang akan ditambahkan |
| **Flow Of Event** | 1. Use case dimulai ketika sistem menampilkan form tambah diskon 2. Admin menginputkan Kode Promo 3. Sistem memvalidasi keunikan kode 4. Admin menginputkan tanggal mulainya promo 5. Admin menginputkan tanggal berakhir promo 6. Sistem memvalidasi tanggal mulai dan berakhir 7. Admin memilih jenis promo Buy x Get y 8. Admin menginput kode produk x 9. Sistem menampilkan nama produk x 10. Admin menginputkan jumlah produk minimum 11. Sistem memvalidasi jumlah 12. Admin menginputkan kode produk y 13. Sistem menampilkan nama produk y 14. Admin menginputkan jumlah produk y 15. Sistem melakukan validasi 16. Sistem menampilkan pesan konfirmasi tambah diskon 17. Admin memilih Ya pada pilihan yang muncul 18. Sistem mencatat data diskon 19. Use case selesai |
| **Post Condition** | Sistem mencatat data promo Buy x Get y |
| **Alternative Flow** | Pada langkah 3, 6, 11, dan 15 jika sistem menilai bahwa inputan tidak valid mana sistem menampilkan pesan kesalahan dan meminta input ulang  Pada langkah nomor 17, jika Admin memilih “Tidak” pada pilihan yang muncul maka sistem membatalkan penambahan data promo dan proses kembali ke langkah 1 |

3. Mengubah Promo

|  |  |
| --- | --- |
| **Nama Use Case** | Mengubah Promo |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin mengubah data promo tertentu |
| **Pre Condition** | Admin berada pada halaman promo memiliki kode promo yang akan diubah |
| **Flow Of Event** | 1. Use case dimulai ketika Admin memilih promo yang akan diubah yang dapat dilakukan dengan melakukan pencarian terlebih dahulu(misal promo Diskon Produk ) 2. Admin mengklik button Ubah 3. Sistem menampilkan form ubah promo beserta data lama 4. Admin dapat mengubah tanggal berakhir, dan persentase diskon 5. Sistem menampilkan pesan konfirmasi update 6. Admin memilih “Ya” pada pilihan yang ada 7. Sistem memperbarui tanggal berakhir dan atau persentase diskon dan menampilkan pesan berhasil 8. Use case selesai |
| **Post Condition** | Sistem memperbarui tanggal berakhir dan atau persentase diskon dan menampilkan pesan berhasil |
| **Alternative Flow** | Pada langkah 6, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses mengubah produk dibatalkan dan proses kembali ke langkah 1  Apabila data yang diubah adalah jenis promo Diskon total belanja maka data yang dapat diubah adalah tanggal berakhir, total belanja minimal, dan persentase diskon  Apabila daya yang diubah adalah jenis promo Buy x get y maka data yang dapat diubah adalah tanggal berakhir, jumlah minumum produk x, jumlah produk y |

4. Menghapus Promo

|  |  |
| --- | --- |
| **Nama Use Case** | Menghapus Promo |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin menghapus data promo tertentu |
| **Pre Condition** | Admin berada pada halaman promo dan memiliki kode promo yang akan dihapus |
| **Flow Of Event** | 1. Use case dimulai ketika Admin memilih data promo yang akan dihapus 2. Admin mengklik button Hapus 3. Sistem memastikan bahwa status promo tidak berlaku 4. Sistem menampilkan pesan konfirmasi hapus 5. Admin memilih “Ya” pada pilihan yang ada 6. Sistem menghapus data promo 7. Use case selesai |
| **Post Condition** | Sistem menghapus data produk |
| **Alternative Flow** | Pada langkah 3, jika status promo yang akan dihapus adalah berlaku maka proses hapus promo gagal dan proses kembali ke langkah 1  Pada langkah 5, jika Admin memilih “Tidak” pada pilihan yang muncul maka proses menghapus promo dibatalkan dan proses kembali ke langkah 1 |

**H. Melihat Laporan**

1. Laporan Transaksi

|  |  |
| --- | --- |
| **Nama Use Case** | Melihat Laporan Transaksi |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melihat dan atau mencetak laporan transaksi |
| **Pre Condition** | Admin memilih Halaman Laporan dan memilih Transaksi |
| **Flow Of Event** | 1. Use case dimulai ketika Admin memilih Periode(Mingguan/Bulanan/Tahunan) laporan 2. Admin memilih periode mingguan 3. Admin menginputkan tanggal Mulai 4. Sistem menghitung dan menampilkan tanggal akhir (7 hari dari tanggal mulai) 5. Sistem melakukan pencarian data transaksi yang terjadi pada kurun waktu sesuai inputan 6. Sistem menampilkan data transaksi periode tersebut pada tabel 7. Admin mengklik baris record untuk melihat detail transaksi tertentu 8. Sistem menampilkan kode produk, nama, dan jumlah pada tabel berdasarkan record transaksi yang di klik Admin 9. Admin mengklik button Export 10. Sistem mengexport hasil laporan pada tabel dalam bentuk dokumen 11. Use case selesai |
| **Post Condition** | Sistem menampilkan hasil laporan berdasarkan pilihan admin dan atau mengexport hasil laporan |
| **Alternative Flow** | Admin dapat melakukan pencarian berdasarkan nomor transaksi jika ia bertujuan untuk mengetahui detail transaksi tertentu atau bukan bertujuan untuk melihat laporan  Pada langkah 2, jika Admin memilih periode Bulanan maka Admin menginputkan Bulan dan tahun, proses dilanjutkan kelangkah 5  Pada langkah 2, jika Admin memilih periode Tahunan maka sistem meminta inputan tahun periode dan proses berlanjut ke langkah 5 |

2. Laporan Statistik Penjualan

|  |  |
| --- | --- |
| **Nama Use Case** | Melihat Laporan Statistik |
| **Aktor** | Admin |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana Admin melihat dan atau mencetak laporan statistik penjualan |
| **Pre Condition** | Admin memilih Halaman Laporan dan memilih Statistik |
| **Flow Of Event** | 1. Use case dimulai ketika Admin memilih Periode(Mingguan/Bulanan/Tahunan) laporan 2. Admin memilih periode mingguan 3. Admin menginputkan tanggal Mulai 4. Sistem menghitung dan menampilkan tanggal akhir (7 hari dari tanggal mulai) 5. Sistem melakukan pencarian data transaksi yang terjadi pada kurun waktu sesuai inputan 6. Sistem menampilkan statistik penjualan dalam bentuk tabel berupa kode produk, nama, jumlah transaksi, dan quantity terjual 7. Sistem juga menampilkan daftar maksimal 10 produk terlaris pada periode tersebut 8. Admin mengklik button Export 9. Sistem mengexport hasil laporan dari kedua tabel ke dalam bentuk dokumen 10. Use case selesai |
| **Post Condition** | Sistem menampilkan hasil laporan statistik penjualan, dan daftar maksimal 10 produ terlaris berdasarkan pilihan admin dan atau mengexport hasil laporan |
| **Alternative Flow** | Admin dapat melakukan pencarian berdasarkan kode produk jika ia bertujuan untuk mengetahui detail statistik produk tertentu  Pada langkah 2, jika Admin memilih periode Bulanan maka Admin menginputkan Bulan dan tahun, proses dilanjutkan kelangkah 5  Pada langkah 2, jika Admin memilih periode Tahunan maka sistem meminta inputan tahun periode dan proses berlanjut ke langkah 5 |