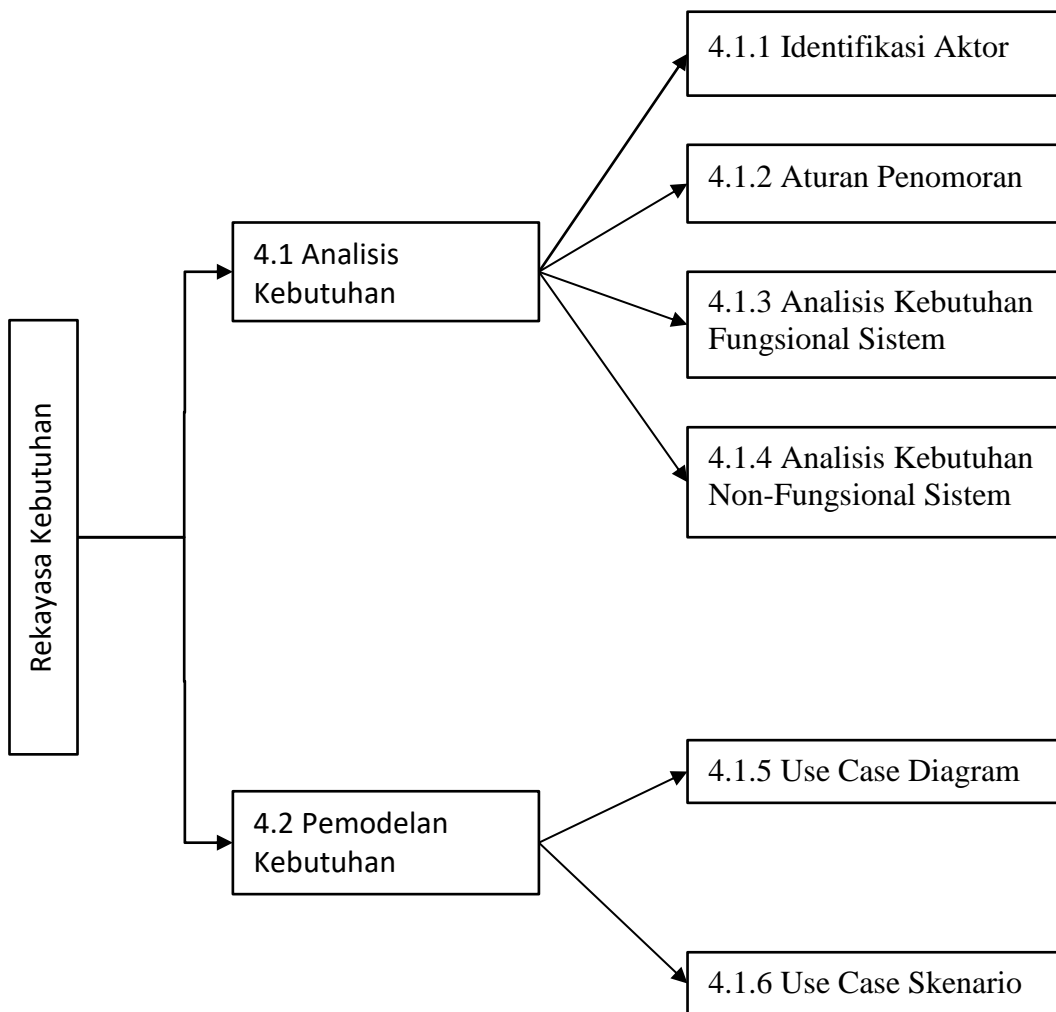


BAB 4 REKAYASA KEBUTUHAN

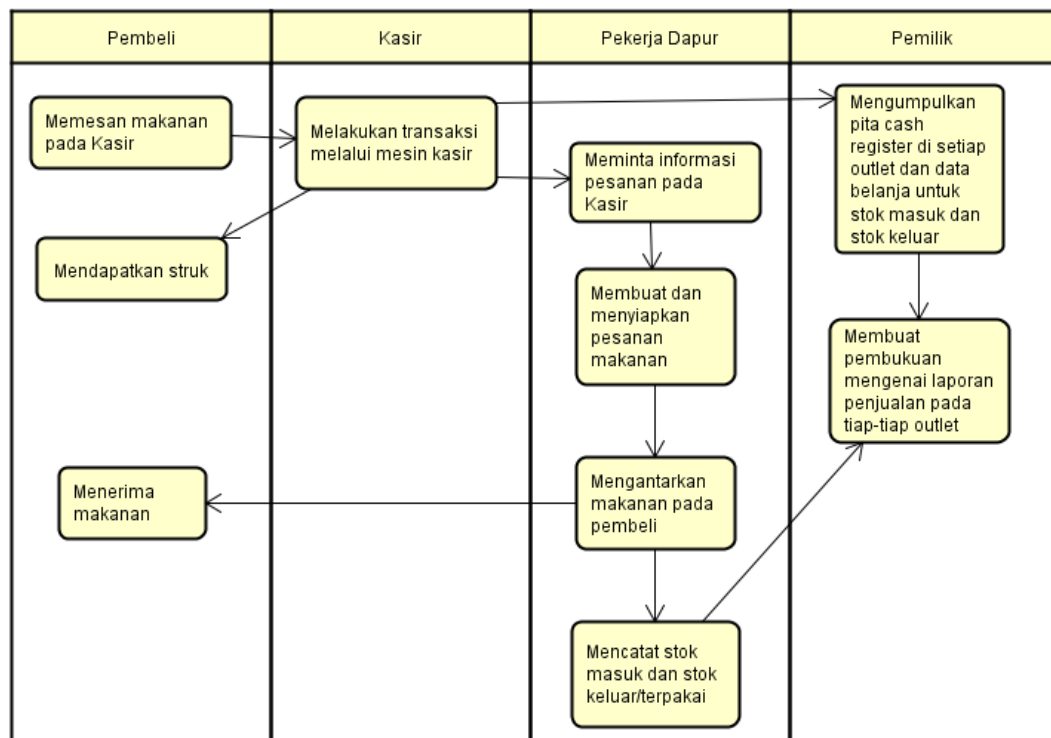
Pada bab ini akan berisi tentang rekayasa kebutuhan yaitu fase yang sangat penting dan krusial dalam membangun suatu sistem. Dimana dalam fase ini, kebutuhan dari pengguna akan dikumpulkan, diulas dan ditetapkan. Hasil yang didapat dari rekayasa kebutuhan ini akan dijadikan acuan untuk melakukan perancangan sistem POS. Pembahasan isi dari bab rekayasa kebutuhan akan ditampilkan dalam gambar 4.1 Diagram pohon rekayasa kebutuhan.



Gambar 0.1 Diagram Pohon Rekayasa Kebutuhan

4.1 Analisis Kebutuhan

Pada tahap analisis kebutuhan bertujuan memberi gambaran tentang kebutuhan-kebutuhan yang harus dipenuhi oleh sistem agar dapat memenuhi kebutuhan pengguna. Sistem yang akan dibangun mengacu pada pengumpulan poin-poin penting dari hasil wawancara yang menunjukkan permasalahan yang sudah dijelaskan pada latar belakang yaitu tidak adanya laporan yang memberitahu keuntungan dari tiap-tiap outlet dan pemberitahuan pada pekerja dapur mengenai pesanan yang harus disiapkan kepada pembeli. Sebelumnya rumah makan pecel pincuk tinuk hanya menggunakan cash register saat melakukan aktivitas transaksi. Namun hal tersebut tidaklah maksimal sehingga dibutuhkan sistem informasi point of sales terintegrasi dalam lingkup rumah makan beserta cabangnya dengan tujuan agar pemilik dapat mengetahui perkembangan aktivitas rumah makannya secara real time dan pemilik tidak harus setiap hari memeriksa ke tiap- tiap outlet untuk meminta *records* transaksi yang sudah dilakukan selama satu hari penuh, tetapi cukup mengakses sistem yang terhubung dengan internet agar mengetahui semua kegiatan yang terjadi di semua cabang dan pada karyawan dapur terdapat fitur untuk melihat pesanan secara realtime secara berurut yang memudahkan dalam proses pembuatan makanan serta ada fitur untuk manajemen stok. Pada Gambar 0.2 Alur **bisnis konvensional pada RM. Pecel Pincuk Bu Tinuk** alur bisnis yang terjadi pada RM Pecel Pincuk Bu Tinuk pada saat belum menerapkan sistem POS terintegrasi antar outlet, sedangkan pada Tabel 0.1 Analisis dan spesifikasi kebutuhan menjelaskan analisis dan spesifikasi kebutuhan untuk membuat sistem POS.



Gambar 0.2 Alur bisnis konvensional pada RM. Pecel Pincuk Bu Tinuk

Tabel 0.1 Analisis dan spesifikasi kebutuhan

Kode	Deskripsi
SAKPL-100	<p>Sistem harus dapat digunakan oleh pemilik dan pekerja rumah makan yang terdiri dari kasir dan dapur.</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus menyediakan halaman untuk login pengguna. (SAKPL-101) - Sistem harus menyediakan field untuk username dan password yang direpresentasikan dengan bentuk persegi panjang. (SAKPL-102)
SAKPL-200	<p>Sistem harus menyediakan sarana untuk mengelola produk.</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus menyediakan halaman untuk menampilkan daftar produk yang sudah pernah dimasukkan oleh Admin. (SAKPL-201) - Sistem harus menyediakan dropdown kategori (semua, makanan, lauk dan minuman) agar dapat memudahkan Admin dalam melihat produk berdasarkan kategori. (SAKPL-202) - Daftar produk yang ditampilkan sistem berupa nama produk, kategori dan harga. (SAKPL-203) - Sistem harus menyediakan tombol aksi edit dan hapus di masing-masing produk yang terdapat pada daftar produk. (SAKPL-204) - Sistem harus menyediakan tombol tambah produk kepada Admin untuk menambah produk baru. (SAKPL-205) - Sistem harus menyediakan form isian untuk melakukan tambah produk kepada Admin. (SAKPL-206) - Pada form untuk menambah produk harus ada field untuk mengisi nama produk, dropdown pilihan kategori, field harga dan radio button untuk pilihan lacak produk atau tidak. (SAKPL-207) - Sistem harus menyediakan tombol tambah pada form tambah produk agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-208)

Kode	Deskripsi
	<ul style="list-style-type: none"> - Sistem harus menyediakan form isian untuk melakukan edit produk yang telah tersimpan kepada Admin. (SAKPL-209) - Sistem harus menyediakan tombol simpan pada form edit produk agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-210)
SAKPL-300	<p>Sistem harus menyediakan sarana untuk mengelola bahan baku.</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus menyediakan halaman untuk menampilkan daftar bahan baku yang sudah pernah dimasukkan oleh Admin. (SAKPL-301) - Daftar produk yang ditampilkan sistem berupa nama bahan baku dan tipe satuannya . (SAKPL-302) - Sistem harus menyediakan tombol aksi edit dan hapus, pada daftar bahan baku. (SAKPL-303) - Sistem harus menyediakan fasilitas untuk mencari bahan baku yang direpresentasikan dengan bentuk persegi panjang dan terdapat label cari. (SAKPL-304) - Sistem harus menyediakan tombol tambah bahan baku kepada Admin untuk menambah bahan baku baru. (SAKPL-305) - Sistem harus menyediakan form isian untuk melakukan tambah bahan baku kepada Admin. (SAKPL-306) - Pada form untuk tambah bahan baku harus ada field untuk mengisi nama bahan baku dan dropdown tipe satuan. (SAKPL-307) - Sistem harus menyediakan tombol tambah pada form tambah bahan baku agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-308) - Sistem harus menyediakan form isian untuk melakukan edit bahan baku yang telah tersimpan kepada Admin. (SAKPL-309) - Sistem harus menyediakan tombol simpan pada form edit bahan baku agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-310)

Kode	Deskripsi
	<ul style="list-style-type: none"> - Sistem harus menyediakan form untuk menambah stok masuk/keluar. (SAKPL-311)
SAKPL-400	<p>Sistem harus menyediakan sarana untuk mengelola data user</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus menyediakan halaman untuk menampilkan daftar user yang sudah pernah dimasukkan oleh Admin. (SAKPL-401) - Data user yang ditampilkan sistem berupa nama, level, username, outlet dan nomor telepon. (SAKPL-402) - Sistem harus menyediakan tombol aksi edit dan hapus di masing-masing user yang terdapat pada daftar user. (SAKPL-403) - Sistem harus menyediakan tombol tambah user kepada Admin untuk menambah user baru. (SAKPL-404) - Sistem harus menyediakan form isian untuk melakukan tambah user kepada Admin. (SAKPL-405) - Pada form untuk menambah produk harus ada field untuk mengisi nama, dropdown pilihan level dan outlet, field username, password dan nomor telepon. (SAKPL-406) - Sistem harus menyediakan tombol tambah pada form tambah user agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-407) - Sistem harus menyediakan form isian untuk melakukan edit user yang telah tersimpan. (SAKPL-408) - Sistem harus menyediakan tombol simpan pada form edit produk agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-409)
SAKPL-500	<p>Sistem harus menyediakan sarana untuk mengelola outlet</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus menyediakan halaman untuk menampilkan daftar outlet yang sudah pernah dimasukkan oleh Admin. (SAKPL-501)

Kode	Deskripsi
	<ul style="list-style-type: none"> - Daftar outlet yang ditampilkan sistem berupa nama, alamat dan nomor telepon. (SAKPL-502) - Sistem harus menyediakan tombol aksi edit dan hapus di masing-masing user yang terdapat pada daftar user. (SAKPL-503) - Sistem harus menyediakan tombol tambah outlet kepada Admin untuk menambah outlet baru. (SAKPL-504) - Sistem harus menyediakan form isian untuk melakukan tambah outlet kepada Admin. (SAKPL-505) - Pada form untuk menambah produk harus ada field untuk mengisi nama, alamat dan nomor telepon. (SAKPL-506) - Sistem harus menyediakan tombol tambah pada form tambah outlet agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-507) - Sistem harus menyediakan form isian untuk melakukan edit outlet yang telah tersimpan. (SAKPL-508) - Sistem harus menyediakan tombol simpan pada form edit produk agar masukan yang telah diisi Admin dapat diproses untuk disimpan. (SAKPL-509)
SAKPL-600	<p>Sistem harus menyediakan sarana untuk melakukan transaksi</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus mampu menyediakan halaman untuk menambah transaksi kepada kasir. (SAKPL-601) - Setiap menu yang dijual harus ditampilkan dengan nama masing-masing produk yang dipisahkan oleh kategori (makanan/lauk/minuman). (SAKPL-602) - Kasir harus disediakan tombol x untuk membatalkan pesanan yang tidak jadi dipesan oleh pembeli. (SAKPL-603) - Sistem harus mampu menampilkan total pembayaran ketika Kasir sudah dalam melakukan transaksi. (SAKPL-604)

Kode	Deskripsi
	<ul style="list-style-type: none"> - Sistem harus mampu menampilkan uang kembalian ketika Kasir sudah memasukkan uang pembayaran yang diberikan oleh pembeli. (SAKPL-605) - Kasir harus disediakan tombol bayar untuk melakukan pembayaran. (SAKPL-606) - Sistem harus menampilkan struk untuk dicetak. (SAKPL-607)
SAKPL-700	<p>Sistem harus menyediakan sarana bagi user dapur untuk mengelola dan menampilkan antrian pesanan.</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus mampu menampilkan urutan pesanan berdasarkan antrian yang lebih dulu memesan. (SAKPL-701) - Sistem harus mampu menampilkan daftar semua menu beserta jumlah kumulatif yang dipesan oleh pembeli. (SAKPL-702) - Sistem harus menyediakan tombol aksi untuk mengurangi jumlah menu yang sudah disiapkan. (SAKPL-703) - Sistem harus menyediakan tombol “Saji” untuk pesanan yang sudah selesai dibuat dan siap diantar ke pembeli. (SAKPL-704) - Sistem harus menyediakan tombol “Selesai” untuk pesanan yang sudah selesai diantar ke pembeli. (SAKPL-705) - Sistem harus mampu menampilkan pesanan yang sudah dibuat. (SAKPL-706)
SAKPL-800	<p>Sistem harus menyediakan sarana bagi user dapur untuk memasukkan stok masuk dan stok keluar bahan.</p> <ul style="list-style-type: none"> - Sistem harus menyediakan halaman untuk menampilkan arsip stok, berisi informasi nama bahan, jumlah stok masuk, jumlah stok keluar dan jumlah stok saat ini. (SAKPL-801) - Sistem harus menyediakan tombol untuk input stok masuk/keluar. (SAKPL-802) - Sistem harus menyediakan form isian untuk melakukan tambah stok masuk/keluar pada user dapur. (SAKPL-803)

Kode	Deskripsi
	<ul style="list-style-type: none"> - Pada form untuk menambah stok masuk harus ada field untuk mengisi jumlah stok yang ditambahkan dan harga. (SAKPL-804) - Pada form untuk menambah stok keluar harus ada field untuk mengisi jumlah stok keluar. (SAKPL-805) - Sistem harus menyediakan tombol tambah pada form tambah stok masuk/keluar agar masukan yang telah diisi dapat disimpan. (SAKPL-806)
SAKPL-900	<p>Sistem harus menyediakan fasilitas bagi Admin agar dapat melihat laporan penjualan, laporan produk dan laporan stok.</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> - Sistem harus mampu menampilkan laporan penjualan harian berdasarkan range tanggal dan pilihan outlet yang dimasukkan oleh Admin. (SAKPL-901) - Data laporan penjualan yang ditampilkan berupa tanggal, outlet, jumlah transaksi dan total penjualan serta total seluruh jumlah transaksi dan total penjualan keseluruhan dari tanggal awal dan akhir. (SAKPL-901) - Sistem harus mampu menampilkan laporan omset dengan memberikan pilihan bulan yang dimasukkan oleh Admin. (SAKPL-901) - Data laporan omset yang ditampilkan berupa tahun, bulan, outlet, total penjualan, total pengeluaran dan laba. (SAKPL-901) - Sistem harus mampu menampilkan laporan riwayat data transaksi berdasarkan tanggal dan outlet yang dipilih oleh Admin. (SAKPL-901) - Data laporan riwayat data transaksi yang ditampilkan berupa kasir, waktu, produk, jumlah, total dan grand total. (SAKPL-901) - Sistem harus mampu menampilkan laporan produk berdasarkan tanggal dan outlet yang dipilih oleh Admin. (SAKPL-901) - Data laporan produk yang ditampilkan nama produk, kategori, jumlah produk terjual dan total nominal. (SAKPL-901)

Kode	Deskripsi
	<ul style="list-style-type: none"> - Sistem harus mampu menampilkan laporan stok berdasarkan range tanggal dan outlet yang dipilih oleh Admin. (SAKPL-901) - Data laporan produk yang ditampilkan berupa nama bahan, stok masuk dan stok keluar. (SAKPL-901)

Pada penelitian ini juga melakukan pengujian non-fungsional berupa pengujian *performance*. Hal demikian merujuk pada penelitian yang dilakukan oleh Scott Barber yang berjudul "How fast does a website need to be?" (Barber, 2006) yang mengatakan bahwa ekspektasi pengguna ketika menyangkut tentang kinerja adalah *response time*. Pengguna tidak peduli tentang berapa banyak pengguna yang mengakses sistem pada satu waktu atau bagaimana sistem dirancang untuk pulih jika terjadi bencana. Merangkum dari apa yang dilakukan Scott bahwa ekspektasi pengguna tentang *loading time* yang cepat adalah dibawah 3 detik sedangkan jika 3-5 detik maka dianggap normal. Kategori lambat bila *loading time* antara 5 hingga 8 detik, apabila *loading time* 8-15 detik maka pengguna akan frustrasi dan yang paling tidak bisa diterima adalah ketika *loading time* lebih dari 15 detik. Pada sistem POS yang akan dibangun harus memiliki *loading time* kurang dari 3 detik agar dapat dikatakan cepat.

4.1.1 Identifikasi Aktor

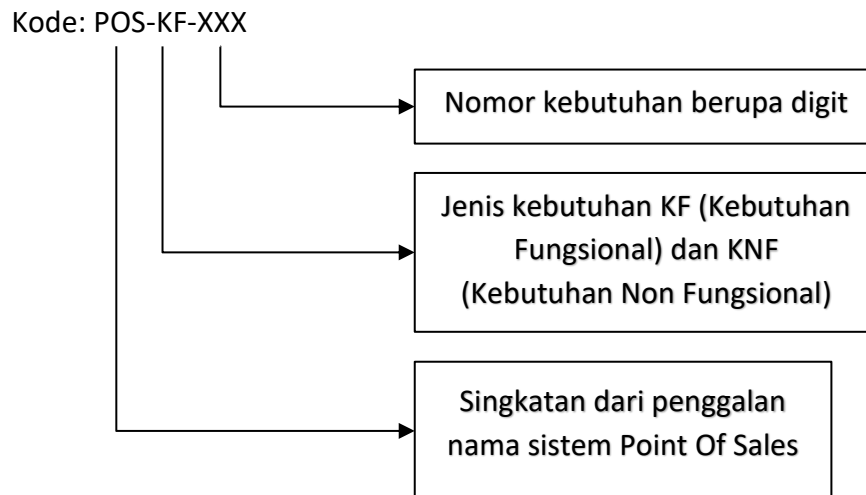
Pada tahap identifikasi aktor ditujukan untuk mengidentifikasi pengguna yang akan menggunakan sistem yang akan dibangun agar pengguna dapat berinteraksi dengan sistem. Pengguna terdiri dari empat kategori, yaitu user yaitu umum, lalu admin adalah pemilik sebagai admin yaitu orang yang dapat melihat seluruh fungsi yang ada pada sistem, pengguna kasir yaitu orang yang dapat melakukan kegiatan transaksi dan yang terakhir pengguna dapur yaitu yang dapat melihat antrian pesanan dan dapat menambah stok masuk/keluar bahan. Daftar aktor beserta deskripsinya akan ditunjukkan pada tabel 4.2 identifikasi aktor.

Tabel 0.2 Identifikasi aktor

No	Aktor	Deskripsi
1	User (U)	Aktor ini belum dapat melakukan akses terhadap sistem hanya sebatas pada tampilan login.
2	Admin (A)	Aktor dapat mengelola semua pengguna dan melihat laporan berupa informasi seputar kegiatan rumah makan.
3	Kasir (K)	Aktor ini dapat melakukan transaksi.
4	Dapur (D)	Aktor ini dapat melihat urutan pesanan makanan yang baru dipesan dan dapat mengelola informasi stok bahan baku makanan.

4.1.2 Aturan Penomoran Kode Kebutuhan Perangkat Lunak

Aturan penomoran kode kebutuhan perangkat lunak dibuat agar lebih memudahkan dalam memberikan nama kebutuhan fungsional. Dengan adanya aturan penomoran ini diharapkan dalam pemberian nama bisa selaras dengan nama sistem yang dibangun. Dalam gambar 4.2 dipaparkan mengenai aturan penomoran kode kebutuhan perangkat lunak.



Gambar 4.2 Aturan Penomoran

4.1.3 Analisis Kebutuhan Fungsional Sistem

Analisis kebutuhan fungsional merupakan proses analisis untuk mengetahui fungsi apa saja yang harus dipenuhi oleh sistem. Pada kebutuhan fungsional berisi tentang proses-proses apa saja yang akan dilakukan oleh sistem. Berikut daftar kebutuhan fungsional untuk admin dan kasir yang harus ada pada sistem dapat dilihat pada tabel 4.3 spesifikasi kebutuhan fungsional sistem.

Tabel 0.3 Spesifikasi kebutuhan fungsional sistem

Kode Fungsional	Kebutuhan Fungsional	Pengguna	Deskripsi
POS-KF-001	Log in	User	Sistem dapat menerima masukan dari Aktor berupa username dan password yang digunakan untuk masuk kedalam sistem.
POS-KF-002	Logout	Admin, Kasir, Dapur	Sistem dapat membawa Aktor keluar dari sistem.
POS-KF-003	Lihat Produk	Admin	Sistem dapat menampilkan daftar produk yang sudah disimpan

Kode Fungsi	Kebutuhan Fungsional	Pengguna	Deskripsi
POS-KF-004	Tambah Produk	Admin	Sistem menyediakan form untuk menambah produk
POS-KF-005	Edit Produk	Admin	Sistem menyediakan form untuk mengubah produk yang ingin diubah oleh Aktor
POS-KF-006	Hapus Produk	Admin	Sistem menyediakan fungsi kepada Aktor untuk menghapus produk
POS-KF-007	Lihat Bahan baku	Admin	Sistem dapat menampilkan daftar bahan baku yang sudah disimpan
POS-KF-008	Tambah Bahan baku	Admin	Sistem menyediakan form untuk menambah bahan baku
POS-KF-009	Edit Bahan baku	Admin	Sistem menyediakan form untuk mengubah bahan baku yang ingin diubah oleh Aktor
POS-KF-010	Hapus Bahan baku	Admin	Sistem menyediakan fungsi kepada Aktor untuk menghapus Bahan Baku
POS-KF-011	Lihat Outlet	Admin	Sistem dapat menampilkan daftar outlet yang sudah disimpan
POS-KF-012	Tambah Outlet	Admin	Sistem menyediakan form untuk menambah outlet
POS-KF-013	Edit Outlet	Admin	Sistem menyediakan form untuk mengubah outlet yang ingin diubah oleh Aktor
POS-KF-014	Hapus Outlet	Admin	Sistem menyediakan fungsi kepada Aktor untuk menghapus outlet
POS-KF-015	Lihat Pengguna	Admin	Sistem dapat menampilkan daftar pengguna yang sudah disimpan

Kode Fungsi	Kebutuhan Fungsional	Pengguna	Deskripsi
POS-KF-016	Tambah Pengguna	Admin	Sistem menyediakan form untuk menambah pengguna
POS-KF-017	Edit Pengguna	Admin	Sistem menyediakan form untuk mengubah pengguna yang ingin diubah oleh Aktor
POS-KF-018	Hapus Pengguna	Admin	Sistem menyediakan fungsi kepada Aktor untuk menghapus pengguna
POS-KF-019	Lihat Pengeluaran	Admin	Sistem dapat menampilkan daftar pengguna yang sudah disimpan
POS-KF-020	Tambah Pengeluaran	Admin	Sistem menyediakan form untuk menambah pengeluaran dengan memilih jenis pengeluaran operasional/penyusutan
POS-KF-021	Hapus Pengeluaran	Admin	Sistem menyediakan fungsi kepada Aktor untuk menghapus pengeluaran
POS-KF-022	Input Stok Masuk	Dapur	Sistem menyediakan form agar Aktor dapat menginput stok masuk
POS-KF-023	Input Stok Keluar	Dapur	Sistem menyediakan form agar Aktor dapat menginput stok keluar
POS-KF-024	Tambah Transaksi	Kasir	Sistem menyediakan fasilitas kepada Aktor untuk melakukan transaksi
POS-KF-025	Cetak struk	Kasir	Sistem menyediakan fungsi untuk mencetak struk setelah melakukan transaksi
POS-KF-026	Lihat Urutan Pesanan berdasarkan Waktu Pesan	Dapur	Sistem dapat menampilkan urutan pesanan makanan berdasarkan waktu pesan. Yang lebih dulu memesan maka diurutan pertama

Kode Fungsi	Kebutuhan Fungsional	Pengguna	Deskripsi
POS-KF-027	Approve Saji	Dapur	Sistem menyediakan fungsi bagi pengguna dapur untuk menyetujui pesanan yang sudah selesai dibuat ke tahap untuk disajikan ke pembeli
POS-KF-028	Approve Selesai	Dapur	Sistem menyediakan fungsi bagi pengguna dapur untuk menyetujui pesanan yang sudah selesai diantar ke pembeli
POS-KF-029	Lihat Laporan Penjualan Harian	Admin	Sistem dapat menampilkan laporan penjualan harian berdasarkan range tanggal dan outlet
POS-KF-030	Lihat Laporan Omset	Admin	Sistem dapat menampilkan laporan omset berdasarkan bulan yang dipilih
POS-KF-031	Lihat Laporan Data Transaksi	Admin	Sistem dapat menampilkan laporan data transaksi berdasarkan tanggal dan outlet
POS-KF-032	Lihat Laporan Produk	Admin	Sistem dapat menampilkan laporan produk berdasarkan tanggal dan outlet
POS-KF-033	Lihat Laporan Stok	Admin	Sistem dapat menampilkan laporan stok berdasarkan tanggal dan outlet

4.1.4 Analisis Kebutuhan Non-Fungsional Sistem

Selanjutnya akan dijelaskan analisis kebutuhan non-fungsional atau sering disebut *quality attributes* yang menentukan bagaimana sistem berjalan sesuai seharusnya. Kebutuhan ini bertujuan untuk meningkatkan kualitas dari sistem yang akan dibuat. Pada tabel 4.4 akan ditunjukkan kebutuhan non-fungsional.

Tabel 0.4 Kebutuhan non-fungsional

Kode Fungsi	Kebutuhan Non-Fungsional	Deskripsi
POS-KNF-01	<i>Performance</i>	Sistem dapat memberikan respon time kurang dari 3 detik

mengenai skenario use case. Pada tabel 4.5 terdapat skenario use case login, login merupakan aktivitas untuk masuk ke dalam sistem agar dapat mengakses sistem dengan cara memasukkan identitas berupa username dan password.

Tabel 0.5 Skenario use case login

POS-KF-001	Use case login
Objective	Memungkinkan aktor untuk masuk kedalam sistem dan menggunakan sistem
Actor	User
Pre-condition	Aktor mengakses sistem
Main flow	<ol style="list-style-type: none"> 1. Sistem menampilkan form login 2. Aktor memasukkan username dan password 3. Aktor menekan tombol login 4. Sistem kemudian melakukan proses login dengan melakukan pengecekan username dan password yang telah dimasukkan oleh Aktor berdasarkan hak aksesnya 5. Jika verifikasi yang di lakukan Aktor berhasil pada saat cek proses login, maka Aktor akan masuk ke halaman utama.
Alternative flow	A.4 Saat Aktor login, username dan password yang dimasukkan tidak terdaftar saat di lakukan verifikasi username dan password maka sistem akan menampilkan pesan kesalahan bahwa "Maaf, Username/Password tidak cocok."
Post-condition	Jika login berhasil dilakukan, maka akan di pindahkan ke halaman utama panel Aktor sesuai hak akses. Jika tidak, maka akan tetap di halaman login sebelumnya yang menandakan Aktor tidak berhasil melakukan proses login

Ketika ada login tentunya adapula logout, logout berfungsi sebagai proses keluarnya pengguna dari sistem dengan memutus akses ke sistem. Pada proses logout hanya bisa dilakukan oleh pengguna yang dapat mengakses sistem. Skenario use case logout akan ditampilkan pada tabel 4.6.

Tabel 0.6 Skenario use case logout

POS-KF-002	Use case Logout
Objective	Operasi untuk keluar dari sistem
Actor	Admin, Kasir, Dapur

Pre-condition	Aktor sudah masuk kedalam sistem
Main flow	<ol style="list-style-type: none"> 1. Aktor pilih menu logout 2. Sistem menjalankan destroy session dan menampilkan halaman awal sistem
Post-condition	Logout berhasil, Aktor berhasil keluar dari panel halaman aktor dan sistem akan menampilkan halaman utama sistem.

Selanjutnya terdapat skenario use case lihat produk dimana use case ini merupakan include dari use case hapus produk. Ketika ingin melakukan aksi menghapus produk maka secara sendirinya akan menampilkan daftar produk terlebih dahulu baru bisa memilih salah satu produk yang ingin dihapus. Pada tabel 4.7 dibahas skenario use case lihat produk.

Tabel 0.7 Skenario use case lihat produk

POS-KF-003	Use case lihat produk
Objective	Operasi untuk melihat daftar produk
Actor	Admin
Pre-condition	Aktor sudah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih menu produk 2. Aktor memilih pilihan kategori produk (semua/ makanan/ lauk/ minuman) 3. Sistem menjalankan proses lihat produk dan menampilkan data berupa nama, kategori, harga jual dan aksi
Post-condition	Admin berhasil melihat produk

Tambah produk bertujuan untuk menambah produk yang ingin ditambahkan. Hanya Admin yang dapat menambah produk. Pada tabel 4.8 dijelaskan skenario use case untuk tambah produk.

Tabel 0.8 Skenario use case tambah produk

POS-KF-004	Use case tambah produk
Objective	Operasi untuk menambah produk
Actor	Admin
Pre-condition	Aktor berada di halaman menu produk
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol tambah produk

	<ol style="list-style-type: none"> 2. Sistem menampilkan halaman form untuk menambah produk 3. Aktor memasukkan data berupa nama produk memilih kategori produk (makanan/lauk/ minuman) dan memasukkan harga jual 4. Aktor memilih tombol tambah 5. Sistem melakukan pengecekan isi form, Jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Aktor menambah produk dan ada <i>field</i> yang belum diisi maka sistem memberi peringatan untuk mengisi <i>field</i> yang belum terisi
Post-condition	Produk baru berhasil ditambahkan dan sistem menampilkan daftar produk.

Pada daftar produk terdapat dua pilihan aksi yaitu untuk mengubah dan menghapus produk. Pada operasi mengubah produk, yang diubah bisa nama produk, kategori produk (makanan/lauk/minuman), harga jual produk dan dapat pula mengubah apakah stok ingin dilacak. Pada tabel 4.9 dijelaskan skenario use case edit produk.

Tabel 0.9 Skenario use case edit produk

POS-KF-005	Use case edit produk
Objective	Operasi untuk mengubah produk
Actor	Admin
Pre-condition	Aktor berada di halaman menu produk
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol edit pada produk yang ingin diubah 2. Sistem menampilkan halaman form untuk mengedit produk 3. Aktor melakukan pengubahan data yang ingin diubah 4. Aktor menekan tombol edit 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Aktor mengedit produk dan ada yang belum diisi maka sistem akan menampilkan kembali halaman form edit produk
Post-condition	Produk berhasil diubah dan sistem menampilkan daftar produk

Selain mengubah produk terdapat pula aksi untuk menghapus produk. Jika ingin menghapus produk maka memilih salah satu produk yang ingin dihapus lalu produk yang dihapus tersebut akan hilang dari database. Yang dapat menghapus produk hanya admin. Pada tabel 4.10 dijelaskan tentang skenario use case hapus produk.

Tabel 0.10 Skenario use case hapus produk

POS-KF-006	Use case hapus produk
Objective	Operasi untuk menghapus produk
Actor	Admin
Pre-condition	Aktor berada di halaman menu produk
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol hapus pada produk yang ingin dihapus 2. Sistem akan memproses hapus produk yang dipilih oleh Aktor
Post-condition	Produk berhasil dihapus

Lalu terdapat skenario use case lihat bahan baku. Use case ini merupakan include dari use case hapus bahan baku. Ketika ingin menghapus bahan baku maka secara sendirinya akan menampilkan daftar bahan baku terlebih dahulu untuk memilih bahan baku mana yang ingin dihapus. Pada tabel 4.11 dibahas skenario use case lihat bahan baku.

Tabel 0.11 Skenario use case lihat bahan baku

POS-KF-007	Use case lihat bahan baku
Objective	Operasi untuk melihat daftar bahan baku
Actor	Admin
Pre-condition	Aktor sudah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih menu bahan baku 2. Sistem menjalankan proses lihat bahan baku dan menampilkan data berupa nama, tipe satuan dan aksi
Post-condition	Aktor berhasil melihat bahan baku

Tujuan dari operasi tambah bahan baku yaitu untuk menambah bahan baku yang ingin ditambahkan. Hanya Admin yang dapat menambah bahan baku. Pada tabel 4.12 dijelaskan skenario use case untuk tambah bahan baku.

Tabel 0.12 Skenario use case tambah bahan baku

POS-KF-008	Use case tambah bahan baku
Objective	Operasi untuk menambah bahan baku
Actor	Admin
Pre-condition	Aktor berada di halaman menu bahan baku
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol tambah bahan baku 2. Sistem menampilkan halaman form untuk menambah bahan baku 3. Aktor memasukkan data berupa nama bahan baku dan satuannya (pcs, kg, gram, liter, lusin) 4. Aktor memilih tombol tambah 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Aktor menambah bahan baku dan ada yang belum diisi maka sistem akan memberi peringatan untuk mengisi <i>field</i> yang belum diisi
Post-condition	Bahan baku baru berhasil ditambahkan dan sistem menampilkan daftar bahan baku.

Dalam daftar bahan baku terdapat dua pilihan aksi yaitu untuk mengubah dan menghapus bahan baku. Pada operasi mengubah bahan baku, yang diubah bisa nama bahan dan satuan. Pada tabel 4.13 dijelaskan skenario use case edit bahan baku.

Tabel 0.13 Skenario use case edit bahan baku

POS-KF-009	Use case edit bahan baku
Objective	Operasi untuk mengubah bahan baku
Actor	Admin
Pre-condition	Aktor berada di halaman menu bahan baku
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol edit pada bahan baku yang ingin diubah 2. Sistem menampilkan halaman form untuk mengedit bahan baku 3. Aktor melakukan pengubahan data yang ingin diubah 4. Aktor menekan tombol edit

	5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Aktor mengedit bahan baku dan ada yang belum diisi maka sistem akan memberi peringatan untuk mengisi <i>field</i> yang belum diisi
Post-condition	Bahan baku berhasil diubah dan sistem menampilkan daftar bahan baku

Pada operasi untuk menghapus bahan baku, jika ingin menghapus bahan baku maka memilih salah satu bahan yang ingin dihapus lalu bahan yang dihapus tersebut akan hilang dari database. Hanya Admin yang dapat menghapus bahan baku. Pada tabel 4.14 dijelaskan tentang skenario use case hapus bahan baku.

Tabel 0.14 Skenario use case hapus bahan baku

POS-KF-010	Use case hapus bahan baku
Objective	Operasi untuk menghapus bahan baku
Actor	Admin
Pre-condition	Aktor berada di halaman menu bahan baku
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol hapus pada bahan baku yang ingin dihapus 2. Sistem akan memproses hapus bahan baku yang dipilih oleh Aktor
Post-condition	Bahan baku berhasil dihapus

Lalu terdapat skenario use case lihat outlet. Use case ini merupakan include dari use case hapus outlet. Ketika ingin menghapus outlet maka secara sendirinya akan menampilkan daftar outlet terlebih dahulu untuk memilih outlet mana yang ingin dihapus. Pada tabel 4.15 dibahas skenario use case lihat outlet.

Tabel 0.15 Skenario use case lihat outlet

POS-KF-011	Use case lihat outlet
Objective	Operasi untuk melihat daftar outlet
Actor	Admin
Pre-condition	Aktor sudah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih menu outlet 2. Sistem menjalankan proses lihat outlet dan menampilkan data berupa nama outlet, alamat, nomor telepon dan aksi

Post-condition	Aktor berhasil melihat daftar outlet
-----------------------	--------------------------------------

Operasi tambah outlet bertujuan untuk menambah outlet baru yang ingin ditambahkan. Hanya Admin yang dapat menambah produk. Pada tabel 4.16 dijelaskan skenario use case untuk tambah outlet.

Tabel 0.16 Skenario use case tambah outlet

POS-KF-012	Use case tambah outlet
Objective	Operasi untuk menambah outlet
Actor	Admin
Pre-condition	Aktor berada di halaman menu outlet
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol tambah outlet 2. Sistem menampilkan halaman form untuk menambah outlet 3. Aktor memasukkan data berupa nama outlet, alamat dan nomor telepon 4. Aktor memilih tombol tambah 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Aktor menambah outlet dan ada yang belum diisi maka sistem akan memberi peringatan untuk mengisi <i>field</i> yang belum terisi
Post-condition	Outlet baru berhasil ditambahkan dan sistem menampilkan daftar outlet

Terdapat dua pilihan aksi pada daftar outlet yaitu untuk mengubah dan menghapus outlet. Pada operasi mengubah outlet, yang diubah bisa nama outlet, alamat dan nomor telepon. Pada tabel 4.17 dijelaskan skenario use case edit outlet.

Tabel 0.17 Skenario use case edit outlet

POS-KF-013	Use case edit outlet
Objective	Operasi untuk mengubah outlet
Actor	Admin
Pre-condition	Aktor berada di halaman menu outlet
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol edit pada outlet yang ingin diubah

	<ol style="list-style-type: none"> 2. Sistem menampilkan halaman form untuk mengedit outlet 3. Aktor melakukan perubahan data yang ingin diubah 4. Aktor menekan tombol edit 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Aktor mengedit outlet dan ada yang belum diisi maka sistem akan menampilkan peringatan agar mengisi <i>field</i> yang belum terisi
Post-condition	Outlet berhasil diubah dan sistem menampilkan daftar outlet

Operasi selain mengubah outlet yaitu terdapat pula aksi untuk menghapus outlet. Jika ingin menghapus outlet maka memilih salah satu outlet yang ingin dihapus lalu outlet yang dihapus tersebut akan hilang dari database. Yang dapat menghapus outlet hanya Admin. Pada tabel 4.18 dijelaskan tentang skenario use case hapus outlet.

Tabel 0.18 Skenario use case hapus outlet

POS-KF-014	Use case hapus outlet
Objective	Operasi untuk menghapus outlet
Actor	Admin
Pre-condition	Aktor berada di halaman menu outlet
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol hapus pada outlet yang ingin dihapus 2. Sistem akan memproses hapus outlet dengan mengambil id yang dipilih oleh Aktor
Post-condition	Outlet berhasil dihapus

Use case lihat pengguna merupakan include dari use case hapus pengguna. Ketika ingin menghapus pengguna maka secara sendirinya akan menampilkan daftar pengguna terlebih dahulu untuk memilih pengguna mana yang ingin dihapus. Pada tabel 4.19 dibahas skenario use case lihat pengguna.

Tabel 0.19 Skenario use case lihat pengguna

POS-KF-015	Use case lihat pengguna
Objective	Operasi untuk melihat daftar pengguna

Actor	Admin
Pre-condition	Aktor sudah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih menu Data User 2. Sistem menjalankan proses lihat daftar pengguna dan menampilkan data berupa nama, level, username, outlet, nomor telepon dan aksi.
Alternative flows	-
Post-condition	Aktor berhasil melihat daftar outlet

Jika ingin menambah pengguna maka terdapat skenario yang harus dijalankan. Pada tabel 4.8 dijelaskan skenario use case untuk tambah pengguna.

Tabel 0.20 Skenario use case tambah pengguna

POS-KF-016	Use case tambah pengguna
Objective	Operasi untuk menambah pengguna
Actor	Admin
Pre-condition	Aktor berada di halaman menu pengguna
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol tambah pengguna 2. Sistem menampilkan halaman form untuk menambah pengguna 3. Aktor memasukkan data berupa nama pengguna, username, password, outlet dan nomor telepon 4. Aktor memilih tombol tambah 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Admin menambah pengguna dan ada yang belum diisi maka sistem memberi peringatan untuk mengisi <i>field</i> yang belum terisi
Post-condition	Pengguna baru berhasil ditambahkan dan sistem menampilkan daftar pengguna.

Operasi edit pengguna merupakan operasi untuk mengubah salah satu atau banyak data dari salah satu pengguna dan mengisi kata sandi yang baru. Pada tabel 4.21 akan dijelaskan mengenai skenario use case edit pengguna.

Tabel 0.21 Skenario use case edit pengguna

POS-KF-017	Use case edit pengguna
-------------------	-------------------------------

Objective	Operasi untuk mengubah pengguna
Actor	Admin
Pre-condition	Aktor berada di halaman menu pengguna
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol edit pada pengguna yang ingin diubah 2. Sistem menampilkan halaman form untuk mengedit pengguna 3. Aktor melakukan pengubahan data yang ingin diubah dan memasukkan password baru 4. Aktor menekan tombol edit 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.5 Pada saat Aktor mengedit pengguna dan ada yang belum diisi maka sistem memberi peringatan untuk mengisi <i>field</i> yang belum diisi
Post-condition	Pengguna berhasil diubah dan sistem menampilkan daftar pengguna

Jika ingin menghapus pengguna maka pilih salah satu pengguna yang ingin dihapus lalu pengguna yang dihapus tersebut akan hilang dari database. Aktor yang dapat menghapus pengguna hanya Admin. Pada tabel 4.22 dijelaskan tentang skenario use case hapus pengguna.

Tabel 0.22 Skenario use case hapus pengguna

POS-KF-018	Use case hapus pengguna
Objective	Operasi untuk menghapus pengguna
Actor	Admin
Pre-condition	Aktor berada di halaman menu pengguna
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol hapus pada pengguna yang ingin dihapus 2. Sistem akan memproses hapus pengguna yang dipilih oleh Aktor
Post-condition	Pengguna berhasil dihapus

Use case lihat pengeluaran adalah include dari use case hapus pengeluaran. Ketika ingin menghapus, maka secara sendirinya akan menampilkan daftar

pengeluaran terlebih dahulu untuk memilih pengeluaran mana yang ingin dihapus. Pada tabel 4.23 dibahas skenario use case lihat pengeluaran.

Tabel 0.23 Skenario use case lihat pengeluaran

POS-KF-019	Use case lihat pengeluaran
Objective	Operasi untuk melihat daftar pengeluaran
Actor	Admin
Pre-condition	Aktor sudah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih menu Pengeluaran 2. Sistem menampilkan proses lihat daftar pengeluaran dan menampilkan data berupa nama, tanggal, outlet dan jenis (penyusutan/operasional)
Alternative flows	-
Post-condition	Aktor berhasil melihat daftar pengeluaran

Use case tambah pengeluaran merupakan include dari use case hapus pengeluaran. Ketika ingin menghapus pengeluaran maka secara sendirinya akan menampilkan daftar pengeluaran terlebih dahulu untuk memilih pengeluaran mana yang ingin dihapus. Pada tabel 4.24 dibahas skenario use case tambah pengeluaran.

Tabel 0.24 Skenario use case tambah pengeluaran

POS-KF-020	Use case tambah pengeluaran
Objective	Operasi untuk menambah pengeluaran
Actor	Admin
Pre-condition	Aktor berada di halaman menu pengeluaran
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol tambah pengeluaran 2. Sistem menampilkan halaman form untuk menambah 3. Aktor memasukkan nama pengeluaran, memilih kategori pengeluaran (operasional/penyusutan), total uang pengeluaran dan outlet. 4. Aktor memilih tombol tambah 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor

Alternative flows	A.5 Pada saat Aktor menambah pengeluaran dan ada yang belum diisi maka sistem akan memberi peringatan bahwa <i>field</i> harus diisi
Post-condition	Pengeluaran berhasil ditambahkan

Jika ingin menghapus pengeluaran maka memilih salah satu pengeluaran yang ingin dihapus lalu pengeluaran yang dihapus tersebut akan hilang dari database. Yang dapat menghapus pengeluaran adalah Admin. Pada tabel 4.25 dijelaskan tentang skenario use case hapus pengeluaran.

Tabel 0.25 Skenario use case hapus pengeluaran

POS-KF-021	Use case hapus pengeluaran
Objective	Operasi untuk menghapus pengeluaran
Actor	Admin
Pre-condition	Aktor berada di halaman menu pengeluaran
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol hapus pada pengeluaran yang ingin dihapus 2. Sistem akan memproses hapus pengeluaran yang dipilih oleh Aktor
Post-condition	Pengeluaran berhasil dihapus

Pada input stok masuk, yang dapat melakukan fungsi adalah pengguna dapur. Pada tabel 4.26 dibahas mengenai skenario use case input stok masuk.

Tabel 0.26 Skenario use case input stok masuk

POS-KF-022	Use case input stok masuk
Objective	Operasi untuk menambah stok masuk
Actor	Dapur
Pre-condition	Aktor berada di halaman menu stok
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol tambah stok masuk pada bahan baku yang ingin ditambah. 2. Sistem menampilkan halaman form untuk menambah stok masuk 3. Aktor memasukkan jumlah stok masuk dan total harga 4. Aktor memilih tombol tambah

	5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.3 Pada saat Aktor menambah stok masuk dan ada <i>field</i> yang belum diisi maka sistem akan memberi peringatan bahwa <i>field</i> harus diisi
Post-condition	Stok masuk berhasil ditambahkan.

Pada input stok keluar, yang dapat melakukan fungsi adalah pengguna dapur. Pada tabel 4.27 dibahas mengenai skenario use case input stok keluar.

Tabel 0.27 Skenario use case input stok keluar

POS-KF-023	Use case input stok keluar
Objective	Operasi untuk menambah stok keluar
Actor	Dapur
Pre-condition	Aktor berada di halaman menu stok
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih tombol tambah stok keluar pada bahan baku yang ingin ditambah. 2. Sistem menampilkan halaman form untuk menambah stok keluar 3. Aktor memasukkan jumlah stok keluar 4. Aktor memilih tombol tambah 5. Sistem melakukan pengecekan isi form, jika berhasil maka sistem menyimpan masukan Aktor
Alternative flows	A.3 Pada saat Aktor tidak mengisi jumlah stok keluar maka sistem akan memberi peringatan bahwa <i>field</i> harus diisi
Post-condition	Stok keluar berhasil ditambahkan.

Operasi tambah transaksi hanya bisa dilakukan oleh Kasir, sebab yang berhubungan langsung dengan pembeli adalah Kasir. Tugas kasir adalah memasukkan produk yang dipesan oleh pembeli. Untuk melakukan tambah transaksi terdapat skenario use case untuk menambah transaksi yang dijelaskan pada tabel 4.28.

Tabel 0.28 Skenario use case tambah transaksi

POS-KF-024	Use case tambah transaksi
Objective	Operasi untuk melakukan transaksi antar Pembeli dan Kasir

Actor	Kasir
Pre-condition	Aktor masuk kedalam sistem
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih kategori produk (makanan, lauk dan minuman) 2. Sistem menampilkan produk berdasarkan kategori yang dipilih oleh aktor 3. Aktor memilih produk yang dipesan oleh pembeli, memasukkan jumlah produk, status pemesanan, nomor meja dan catatan (jika ada) 4. Sistem menampilkan total pembayaran 5. Aktor memasukkan uang pembayaran dari pembeli 6. Sistem menampilkan jumlah uang kembalian 7. Aktor menekan tombol simpan 8. Aktor memberikan uang kembalian (jika ada) kepada pembeli
Alternative flows	A.3.5 <i>Field</i> ada yang tidak terisi (kecuali catatan), maka sistem memberi peringatan untuk mengisi <i>field</i>
Post-condition	Jika transaksi selesai maka menampilkan halaman transaksi baru

Pada saat kasir sudah melakukan transaksi dan menyimpan transaksi maka sistem akan mengeluarkan struk sebagai bukti pembayaran. Dalam gambar 4.29 ditunjukkan skenario use case cetak struk.

Tabel 0.29 Skenario use case cetak struk

POS-KF-025	Use case cetak struk
Objective	Operasi untuk mencetak struk
Actor	Kasir
Pre-condition	Aktor melakukan transaksi
Main flow	<ol style="list-style-type: none"> 1. Sistem menampilkan struk 2. Aktor menekan tombol cetak
Alternative flows	A.2 Jika aktor tidak menampilkan tombol cetak, melainkan tombol close maka struk tidak akan dicetak.
Post-condition	Sistem akan mencetak struk penjualan

Use case lihat urutan pesanan merupakan include dari use case approve saji. Pada tabel 4.30 akan dibahas tentang skenario use case lihat urutan pesanan.

Tabel 0.30 Skenario use case lihat urutan pesanan

POS-KF-026	Use case lihat urutan pesanan
Objective	Operasi untuk melihat urutan pesanan yang masuk
Actor	Dapur
Pre-condition	Aktor mengakses sistem
Main flow	<ol style="list-style-type: none"> 1. Aktor melakukan login 2. Sistem menampilkan halaman dapur
Post-condition	Sistem menampilkan antrian pesanan secara berurutan

Aktor Dapur dapat melakukan approve saji jika pesanan sudah dibuat. Approve saji ini berguna agar tidak terjadinya penumpukan pada antrian pesanan. Ketika sudah ada pesanan yang sudah jadi disiapkan maka pengguna Dapur bisa langsung menyajikan pesanan dengan menekan tombol saji pada antrian. Pada tabel 4.31 dijelaskan bagaimana skenario use case approve saji.

Tabel 0.31 Skenario use case approve saji

POS-KF-027	Use case approve saji
Objective	Operasi untuk menyetujui pesanan ke tahap antar
Actor	Dapur
Pre-condition	Aktor telah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Sistem menampilkan urutan pesanan dan daftar produk yang dipesan 2. Aktor menekan tombol minus pada produk yang sudah dibuat 3. Sistem menampilkan tombol saji berwarna biru 4. Aktor menekan tombol saji pada pesanan 5. Sistem menampilkan peringatan berupa “apakah sudah yakin ingin menyajikan pesanan? Ya/tidak” 6. Aktor memilih ya. 7. Pesanan yang sudah selesai maka masuk ke daftar makanan yang siap disajikan.
Alternative flows	A.6 pada saat aktor memilih tidak, maka akan kembali ke tampilan daftar antrian pesanan

Post-condition	Pesanan yang sudah di approve akan menghilang dari antrian masak dan berpindah ke antrian antar
-----------------------	---

Setelah pesanan selesai disajikan maka pengguna dapur menekan tombol selesai menandakan jika pesanan sudah selesai diantar. Pada tabel 4.32 dijelaskan bagaimana skenario use case approve selesai.

Tabel 0.32 Skenario use case approve selesai

POS-KF-028	Use case approve selesai
Objective	Operasi untuk menyetujui pesanan ke tahap antar
Actor	Dapur
Pre-condition	Aktor telah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol selesai pada pesanan 2. Sistem menampilkan peringatan berupa “apakah anda pesanan ini telah selesai? Ya/tidak” 3. Aktor memilih ya. 4. Pesanan yang sudah selesai diantar maka masuk ke daftar pesanan selesai.
Alternative flows	A.6 pada saat aktor memilih tidak, maka akan kembali ke tampilan daftar antrian saji
Post-condition	Pesanan yang sudah di approve akan menghilang dari antrian saji dan berpindah ke antrian selesai

Admin dapat melihat laporan penjualan harian dengan range tanggal tertentu. Pada tabel 4.33 merupakan pembahasan dari skenario use case lihat laporan penjualan harian.

Tabel 0.33 Skenario use case lihat laporan penjualan harian

POS-KF-029	Use case lihat laporan penjualah harian
Objective	Operasi untuk melihat laporan penjualan harian
Actor	Admin
Pre-condition	Aktor telah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol laporan dan memilih pada tombol laporan penjualan 2. Sistem menampilkan halaman laporan penjualan 3. Aktor memilih tombol penjualan harian 4. Sistem menampilkan pilihan range tanggal dan outlet

	5. Aktor memasukkan tanggal awal, tanggal akhir dan outlet 6. Aktor menekan tombol lihat 7. Sistem menampilkan data berupa tanggal, outlet, jumlah transaksi dan total penjualan
Post-condition	Aktor dapat melihat laporan penjualan harian

Pada tabel 4.34 membahas tentang skenario use case lihat laporan omset. Laporan omset yang dapat dilihat adalah informasi mengenai laba dari tiap-tiap outlet maupun penggabungan dari semua outlet.

Tabel 0.34 Skenario use case lihat laporan omset

POS-KF-030	Use Use case lihat laporan omset
Objective	Operasi untuk melihat laporan omset
Actor	Admin
Pre-condition	Aktor telah melakukan login
Main flow	1. Aktor menekan tombol laporan dan memilih pada tombol laporan penjualan 2. Sistem menampilkan halaman laporan penjualan 3. Aktor memilih tombol omset 4. Sistem menampilkan pilihan bulan 5. Aktor memilih bulan yang ingin dilihat 6. Aktor menekan tombol lihat 7. Sistem menampilkan data berupa tahun, bulan, outlet, total penjualan, total pengeluaran dan laba
Post-condition	Aktor dapat melihat laporan omset dalam satu bulan atau satu tahun

Laporan data transaksi bertujuan untuk menampilkan riwayat data transaksi yang terjadi pada satu hari penuh. Pada tabel 4.35 akan membahas tentang skenario use case lihat laporan data transaksi.

Tabel 0.35 Skenario use case lihat laporan data transaksi

POS-KF-031	Use case lihat laporan data transaksi
Objective	Operasi untuk melihat laporan riwayat data transaksi
Actor	Admin

Pre-condition	Aktor telah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol laporan dan memilih pada tombol laporan penjualan 2. Sistem menampilkan halaman laporan penjualan 3. Aktor memilih tombol data transaksi penjualan 4. Sistem menampilkan pilihan tanggal dan outlet 5. Aktor memasukkan tanggal dan outlet 6. Aktor menekan tombol lihat 7. Sistem menampilkan data berupa kasir, waktu, produk, jumlah, total dan grand total
Post-condition	Aktor dapat melihat laporan data transaksi penjualan

Admin dapat melihat laporan produk dengan cara memasukkan range tanggal awal dan akhir. Laporan produk berfungsi menampilkan data penjualan pada masing-masing produk pada tanggal yang telah dimasukkan oleh Admin. Pada tabel 4.36 membahas mengenai skenario use case lihat laporan produk.

Tabel 0.36 Skenario use case lihat laporan produk

POS-KF-032	Use case lihat laporan produk
Objective	Operasi untuk melihat laporan produk
Actor	Admin
Pre-condition	Aktor telah melakukan login
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol laporan dan memilih pada tombol laporan produk 2. Sistem menampilkan pilihan tanggal dan outlet 3. Aktor memasukkan tanggal dan outlet 4. Aktor menekan tombol lihat 5. Sistem menampilkan data berupa nama produk, kategori, jumlah terjual dan total nominal
Post-condition	Aktor dapat melihat laporan produk

Laporan stok bertujuan untuk menampilkan stok masuk dan stok keluar yang sudah terjadi dalam beberapa jarak tanggal yang sudah ditentukan oleh Admin. Pada tabel 4.37 menjelaskan tentang skenario use case lihat laporan stok.

Tabel 0.37 Skenario use case lihat laporan stok

POS-KF-033	Use case laporan stok
Objective	Operasi untuk melihat laporan stok
Actor	Admin
Pre-condition	Aktor telah melakukan login
Main flow	<ol style="list-style-type: none">1. Aktor menekan tombol laporan dan memilih pada tombol laporan stok2. Sistem menampilkan pilihan tanggal dan outlet3. Aktor memasukkan tanggal dan outlet4. Aktor menekan tombol lihat5. Sistem menampilkan data berupa nama bahan, stok masuk dan stok keluar
Post-condition	Aktor dapat melihat laporan stok

