

---

# **Software Requirements Specification**

**untuk**

**Sistem Manajemen Stok untuk Jeje Katering  
(J Stok)**

**Versi 2025.0.1**

**Oleh :**

**Safdar Rahman (23500811349)**

**Yuda Kurniawan (2350081143)**

**Puja Dwi (2350081105)**

**Yurika Tiyyara(235008117)**

**30 November 2025**

## Daftar Isi

1.	Pendahuluan .....	4
1.1	Tujuan .....	4
1.2	Deskripsi Umum Dokumen .....	4
1.3	Ruang Lingkup Produk .....	4
1.4	Referensi .....	5
2.	Deskripsi Umum .....	5
2.1	Perspektif Produk .....	5
2.2	Karakteristik Pengguna .....	5
2.3	Batasan Umum .....	6
3.	Kebutuhan Spesifik .....	6
•	FU-02: Peringatan Stok Minimum .....	6
•	FU-03: Pelaporan .....	6
•	FU-04: Jejak Audit (Audit Trail) .....	6
•	FU-05: Autentikasi dan Otorisasi .....	6

3.2	Kebutuhan Non-Fungsional .....	7
3.2.2	Kebutuhan Keamanan.....	7
3.2.3	Atribut Kualitas Perangkat Lunak .....	7
3.3	Kebutuhan Antarmuka Eksternal .....	7
3.3.2	Antarmuka Perangkat Lunak .....	7
3.3.3	Antarmuka Komunikasi.....	8

## 1. Pendahuluan

### 1.1 Tujuan

Dokumen ini merupakan Spesifikasi Kebutuhan Perangkat Lunak (SKPL) atau *Software Requirements Specification* (SRS) untuk "Sistem Manajemen Stok untuk Jeje Katering". Tujuan dari penyusunan dokumen ini adalah untuk memberikan penjelasan rinci mengenai perangkat lunak yang akan dikembangkan, mencakup gambaran umum, fitur, batasan, serta kebutuhan fungsional dan non-fungsional.

Dokumen ini ditujukan untuk menjadi referensi teknis utama bagi tim pengembang dalam proses perancangan, pengembangan, dan pengujian sistem. Selain itu, dokumen ini juga digunakan oleh pemilik usaha dan pihak terkait sebagai acuan untuk memastikan bahwa sistem yang dibangun sesuai dengan tujuan dan kebutuhan bisnis Jeje Katering, serta sebagai bagian dari pemenuhan tugas mata kuliah Rekayasa Perangkat Lunak.

### 1.2 Deskripsi Umum Dokumen

Dokumen ini menguraikan spesifikasi kebutuhan untuk sistem manajemen stok.

- **Bab 1 (Pendahuluan):** Menjelaskan tujuan pembuatan dokumen, ruang lingkup produk, dan referensi yang digunakan.
- **Bab 2 (Deskripsi Umum):** Memberikan gambaran umum produk, karakteristik pengguna yang akan mengoperasikan sistem, serta batasan-batasan umum dalam pengembangan.
- **Bab 3 (Kebutuhan Spesifik):** Merinci kebutuhan fungsional (apa yang bisa dilakukan sistem), non-fungsional (kriteria teknis seperti performa dan keamanan), serta kebutuhan antarmuka.
- **Bab 4 (Lampiran):** Berisi informasi tambahan seperti ketentuan dukungan dan pemeliharaan.

### 1.3 Ruang Lingkup Produk

Perangkat lunak yang akan dikembangkan adalah sebuah sistem manajemen stok berbasis web yang dirancang khusus untuk Jeje Katering. Jeje Katering merupakan sebuah CV yang telah beroperasi selama 27 tahun dan melayani klien pabrik berskala nasional dengan rata-rata 300 pesanan per hari.

Saat ini, proses pencatatan stok bahan makanan masih dilakukan secara manual, yang sering mengakibatkan masalah kelebihan atau kekurangan stok. Sistem ini bertujuan untuk mengatasi masalah tersebut dengan menyediakan platform digital untuk:

- Mencatat pemasukan dan pengeluaran bahan makanan.
- Memberikan peringatan stok.
- Menghasilkan laporan stok.

Tujuan utamanya adalah meningkatkan efisiensi operasional, mengurangi risiko kesalahan stok, dan menjadi fondasi untuk transformasi digital perusahaan di masa depan.

## 1.4 Referensi

- Dokumen Kerangka Acuan Kerja (KAK) Teknis Untuk Pengembangan Sistem Manajemen Stok untuk Jeje Katering, Versi 2025.0.1.

## 2. Deskripsi Umum

### 2.1 Perspektif Produk

Sistem ini merupakan sistem baru yang berdiri sendiri (stand-alone), dirancang untuk menggantikan proses pencatatan stok manual yang ada saat ini. Sistem akan dibangun dengan arsitektur modular yang memungkinkan pengembangan dan penambahan fitur di kemudian hari.

### 2.2 Karakteristik Pengguna

Pengguna sistem ini adalah staf internal Jeje Katering yang memiliki tingkat keahlian teknis yang bervariasi. Oleh karena itu, sistem harus dirancang agar intuitif dan mudah digunakan.

Tipe Pengguna	Definisi dan Tugas Utama
<b>Admin (Pemilik Usaha)</b>	Memiliki akses penuh ke semua fitur sistem, termasuk manajemen pengguna, melihat laporan, dan konfigurasi sistem.
<b>Pegawai Stok</b>	Bertugas untuk melakukan pencatatan transaksi harian (pemasukan dan pengeluaran bahan makanan) dan melihat status stok saat ini.

*Catatan: Hak akses spesifik untuk setiap peran akan didefinisikan lebih lanjut pada tahap perancangan.*

## 2.3 Batasan Umum

- Sistem dikembangkan menggunakan model *Prototyping*.
- Tim pengembang minimal terdiri dari 3 *developer* dan 1 *UI designer*.
- Kode sumber (*source code*) yang dihasilkan akan menjadi milik penuh Jeje Katering.
- Vendor tidak diperbolehkan mengalihkan pekerjaan kepada pihak ketiga tanpa persetujuan dari Jeje Katering.

## 3. Kebutuhan Spesifik

### 3.1 Kebutuhan Fungsional

- **FU-01: Manajemen Stok Bahan Makanan**
  - Sistem harus dapat mencatat data bahan makanan baru, termasuk nama bahan dan satuan.
  - Sistem harus dapat mencatat transaksi pemasukan bahan makanan (stok masuk).
  - Sistem harus dapat mencatat transaksi pengeluaran bahan makanan (stok keluar) untuk kebutuhan produksi harian.
- **FU-02: Peringatan Stok Minimum**
  - Sistem harus dapat memberikan peringatan atau notifikasi visual ketika jumlah stok suatu bahan makanan telah mencapai batas minimum yang ditentukan.
- **FU-03: Pelaporan**
  - Sistem harus dapat menghasilkan laporan stok sederhana yang dapat diekspor ke format Excel atau CSV.
- **FU-04: Jejak Audit (Audit Trail)**
  - Sistem harus dapat mencatat riwayat setiap perubahan data stok (penambahan, pengurangan), termasuk informasi waktu dan pengguna yang melakukan perubahan.
- **FU-05: Autentikasi dan Otorisasi**
  - Sistem harus menyediakan halaman login untuk membatasi akses.
  - Sistem harus dapat mengimplementasikan kontrol akses berbasis peran (*role-based access*) antara Admin dan Pegawai Stok.

## 3.2 Kebutuhan Non-Fungsional

### 3.2.1 Kebutuhan Performansi

- Sistem harus ringan dan dapat diakses dengan cepat bahkan pada perangkat dengan spesifikasi rendah.

### 3.2.2 Kebutuhan Keamanan

- Sistem harus memiliki mekanisme autentikasi pengguna (login dengan username dan password) untuk mencegah akses yang tidak sah.
- Proses *backup* data harus dapat dilakukan secara manual oleh pengguna (misalnya mingguan), dengan panduan yang disediakan oleh vendor.

### 3.2.3 Atribut Kualitas Perangkat Lunak

- **Kemudahan Penggunaan (*Usability*):** Antarmuka pengguna harus intuitif, dan mudah dioperasikan oleh staf non-teknis.
- **Keandalan (*Reliability*):** Sistem harus dapat diandalkan untuk operasional harian tanpa sering mengalami *crash* atau *error*.
- **Skalabilitas (*Scalability*):** Arsitektur sistem harus bersifat modular untuk mempermudah penambahan fitur di masa depan.
- **Pemeliharaan (*Maintainability*):** Kode program harus menggunakan konvensi penamaan yang sederhana dan konsisten, serta dilengkapi dokumentasi minimal untuk kemudahan pemeliharaan.

## 3.3 Kebutuhan Antarmuka Eksternal

### 3.3.1 Antarmuka Pengguna

- Sistem akan memiliki antarmuka pengguna grafis Desktop GUI yang dibangun menggunakan teknologi Java dengan Swing atau JavaFX.
- Desain antarmuka akan menerapkan prinsip User-Friendly dengan tata letak formulir dan tabel yang standar pada sistem operasi Windows/Linux/macOS.
- Sistem berjalan sebagai aplikasi executable dan tidak memerlukan browser.

### 3.3.2 Antarmuka Perangkat Lunak

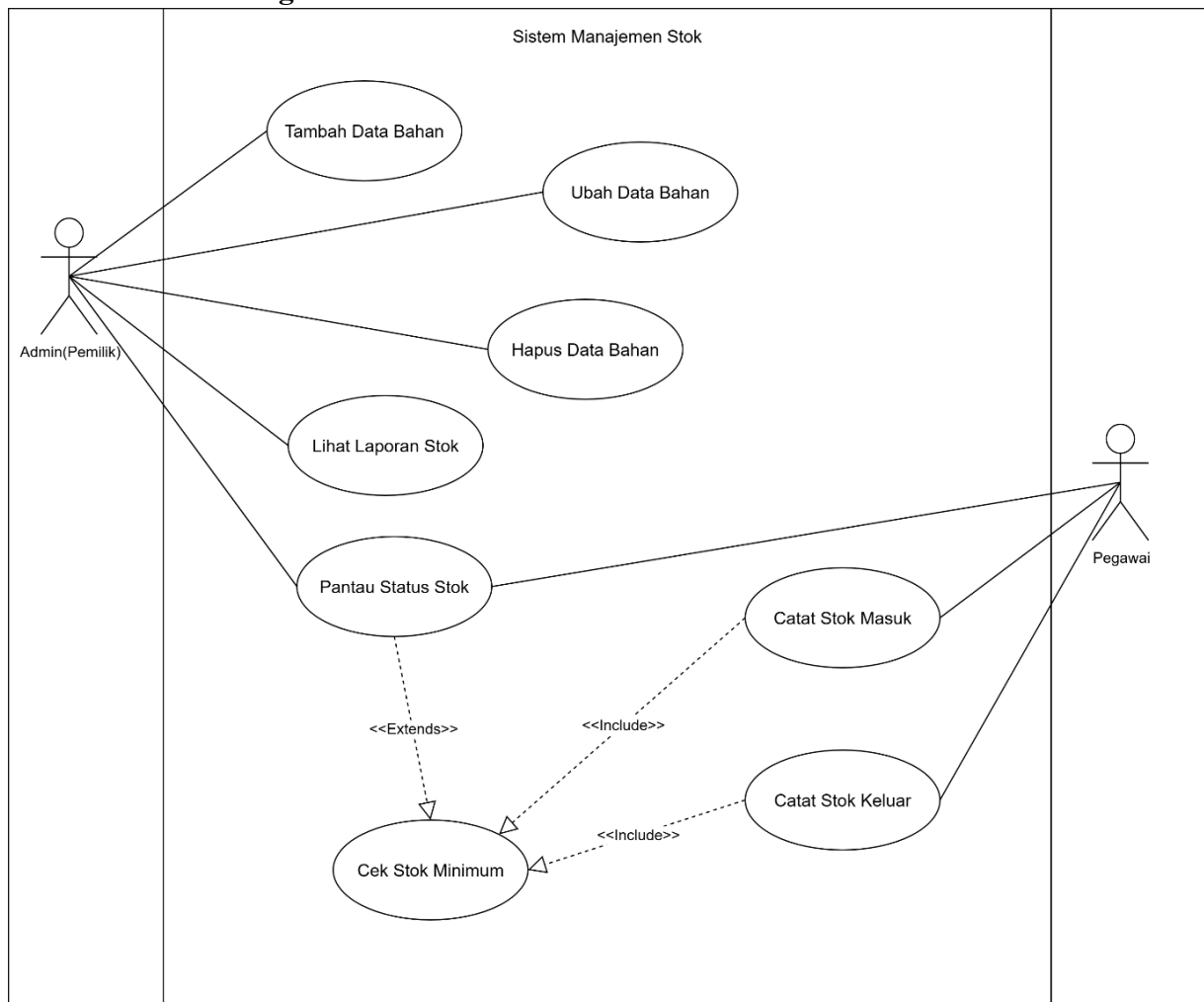
- Bahasa Pemrograman: Java (Java SE Development Kit).
- Database: MySQL.

- Driver Database: JDBC Java Database Connectivity.
- Library GUI: Java Swing atau JavaFX.
- Library Laporan: Apache POI untuk ekspor Excel.

### 3.3.3 Antarmuka Komunikasi

- Aplikasi berkomunikasi langsung dengan server database MySQL menggunakan protokol TCP/IP melalui koneksi JDBC.
- Database diletakkan di komputer yang sama localhost, tidak memerlukan koneksi internet.

### 3.4 Use Case Diagram



**Gambar 1 : Usecase Diagram**



### 3.4.1 Use Case Scenario

#### 3.4.1.1 Use Case: Tambah Data Bahan

<b>Nama Use Case</b>	Tambah Data Bahan
<b>Aktor</b>	Admin (Pemilik)
<b>Deskripsi Singkat</b>	Admin (Pemilik) dapat melakukan penambahan data bahan baku baru ke dalam sistem.
<b>Pre-kondisi</b>	Admin telah login ke dalam sistem.
<b>Skenario Utama (Alur Sukses)</b>	<ol style="list-style-type: none"><li>1. Admin memilih opsi atau menekan tombol "Tambah Data Bahan Baru".</li><li>2. Sistem menampilkan formulir untuk data bahan baru.</li><li>3. Admin mengisi field yang wajib diisi, minimal:<ul style="list-style-type: none"><li>• Nama Bahan (Contoh: "Tepung Terigu")</li><li>• Satuan Dasar (Contoh: "kg", "pcs", "liter")</li><li>• Batas Stok Minimum (Contoh: 10)</li></ul></li><li>4. Admin menekan tombol "Simpan".</li><li>5. Admin mengisi formulir dan menekan tombol "Simpan".</li><li>6. Sistem memvalidasi data, menyimpan bahan baru ke dalam database, dan menampilkan pesan konfirmasi "Data berhasil disimpan".</li><li>7. Daftar bahan makanan pada halaman utama diperbarui dengan data yang baru ditambahkan.</li></ol>

<p><b>Skenario Alternatif</b></p>	<p><b>A. Data Tidak Lengkap:</b></p> <p>3a. Pada formular penambahan data bahan baru, Admin tidak mengisi salah satu field wajib (misal: Satuan).</p> <p>4a. Sistem menampilkan pesan error di samping field tersebut (Contoh: "Satuan tidak boleh kosong").</p> <p>5a. Sistem tetap berada di halaman formulir dan data tidak tersimpan.</p> <p><b>B. Nama Bahan Duplikat:</b></p> <p>3b. Pada formular penambahan data bahan baru, Admin memasukkan "Nama Bahan" yang sudah ada di database.</p> <p>4b. Sistem menampilkan pesan error (Contoh: "Nama bahan 'Tepung Terigu' sudah terdaftar").</p> <p>5b. Sistem tetap berada di halaman formulir.</p> <p><b>C. Admin Membatalkan Penambahan:</b></p> <p>3c. Pada formular penambahan data bahan baru, Admin menekan tombol "Batal".</p> <p>4c. Sistem mengabaikan semua data yang telah diisi.</p> <p>5c. Sistem kembali ke halaman daftar data bahan.</p>
<p><b>Post-kondisi</b></p>	<p>1. Alur sukses: Data bahan baku baru tersimpan di database dan siap digunakan untuk transaksi stok.</p> <p>2. Alur batal: Tidak ada data baru yang tersimpan di sistem.</p>

#### 3.4.1.2 Use Case: Ubah Data Bahan

<p><b>Nama Use Case</b></p>	<p>Ubah Data Bahan</p>
<p><b>Aktor</b></p>	<p>Admin (Pemilik)</p>
<p><b>Deskripsi Singkat</b></p>	<p>Admin dapat melakukan perubahan (mengubah atau memperbarui) detail dari data bahan yang telah terdaftar di sistem.</p>

<b>Pre-kondisi</b>	<ol style="list-style-type: none"> <li>1. Admin telah login ke dalam sistem.</li> <li>2. Minimal harus sudah ada satu data bahan yang terdaftar di sistem.</li> </ol>
<b>Skenario Utama (Alur Sukses)</b>	<ol style="list-style-type: none"> <li>1. Admin memilih salah satu data bahan dari daftar.</li> <li>2. Admin menekan tombol "Edit" untuk data bahan tersebut.</li> <li>3. Sistem menampilkan formulir ubah data yang berisi data saat ini dari bahan tersebut.</li> <li>4. Admin melakukan perubahan pada satu atau lebih field (Contoh: mengubah "Batas Stok Minimum" dari 10 menjadi 15).</li> <li>5. Admin menekan tombol "Simpan Perubahan".</li> <li>6. Sistem melakukan validasi data</li> <li>7. Sistem memperbarui data bahan tersebut di database.</li> <li>8. Sistem menampilkan pesan konfirmasi "Data bahan berhasil diperbarui".</li> <li>9. Sistem kembali ke halaman daftar bahan dan menampilkan data yang telah diperbarui.</li> </ol>
<b>Skenario Alternatif</b>	<p><b>A. Data Tidak Valid:</b></p> <ol style="list-style-type: none"> <li>3a. Pada tampilan formulir perubahan data, Admin menghapus isi field wajib (misal: mengosongkan "Nama Bahan").</li> <li>4a. Sistem menampilkan pesan error "Nama Bahan tidak boleh kosong".</li> <li>5a. Sistem tetap berada di halaman formulir ubah data.</li> </ol> <p><b>B. Admin Membatalkan Perubahan:</b></p> <ol style="list-style-type: none"> <li>3b. Pada tampilan formulir perubahan data, Admin menekan tombol "Batal".</li> <li>4b. Sistem mengabaikan semua perubahan yang telah dibuat.</li> <li>5b. Sistem kembali ke halaman daftar data bahan.</li> </ol>
<b>Post-kondisi</b>	<ol style="list-style-type: none"> <li>1. Alur sukses: Data bahan di database telah diperbarui sesuai dengan input Admin.</li> <li>2. Alur batal: Data bahan di database tetap seperti semula (tidak</li> </ol>

	berubah).
--	-----------

#### 3.4.1.3 Use Case: Hapus Data Bahan

<b>Nama Use Case</b>	Hapus Data Bahan
<b>Aktor</b>	Admin (Pemilik)
<b>Deskripsi Singkat</b>	Admin dapat menghapus data bahan yang telah terdaftar di sistem.
<b>Pre-kondisi</b>	<ol style="list-style-type: none"> <li>1. Admin telah login ke dalam sistem.</li> <li>2. Data bahan yang akan dihapus sudah terdaftar di sistem.</li> </ol>
<b>Skenario Utama (Alur Sukses)</b>	<ol style="list-style-type: none"> <li>1. Admin memilih salah satu data bahan dari daftar.</li> <li>2. Admin menekan tombol "Hapus" untuk data bahan tersebut.</li> <li>3. Sistem memeriksa apakah data bahan tersebut memiliki keterkaitan dengan data lain (Contoh: riwayat transaksi stok).</li> <li>4. Sistem menampilkan dialog konfirmasi penghapusan data bahan.</li> <li>5. Admin memilih "Ya" untuk melakukan penghapusan.</li> <li>6. Sistem memeriksa apakah data bahan tersebut memiliki keterkaitan dengan data lain (Contoh: riwayat transaksi stok).</li> <li>7. Sistem menghapus data bahan dari database.</li> <li>8. Sistem menampilkan pesan konfirmasi "Data bahan berhasil dihapus".</li> <li>9. Sistem memperbarui halaman daftar bahan.</li> </ol>

<p><b>Skenario Alternatif</b></p>	<p><b>A. Admin Membatalkan Penghapusan:</b></p> <p>5a. Admin memilih "Tidak" atau "Batal" pada dialog konfirmasi.</p> <p>6a. Sistem membatalkan operasi hapus.</p> <p>7a. Sistem kembali ke halaman daftar data bahan tanpa ada perubahan.</p> <p><b>B. Data Tidak Bisa Dihapus (Keterkaitan dengan Data penting):</b></p> <p>4b. Sistem mendeteksi bahwa bahan tersebut sudah pernah digunakan dalam transaksi "Stok Masuk" atau "Stok Keluar".</p> <p>5b. Sistem <i>menolak</i> untuk menghapus data (untuk menjaga integritas data laporan).</p> <p>6b. Sistem menampilkan pesan error (Contoh: “Data ‘Tepung Terigu’ tidak dapat dihapus”).</p> <p>7b. Sistem kembali ke halaman daftar data bahan.</p>
<p><b>Post-kondisi</b></p>	<ol style="list-style-type: none"> <li>1. Alur sukses: Data bahan baku tersebut terhapus permanen dari sistem.</li> <li>2. Alur batal/gagal: Data bahan tetap ada di sistem.</li> </ol>

#### 3.4.1.4 Use Case: Catat Stok Masuk

<p><b>Nama Use Case</b></p>	<p>Catat Stok Masuk</p>
<p><b>Aktor</b></p>	<p>Pegawai Katering</p>
<p><b>Deskripsi Singkat</b></p>	<p>Pegawai mencatat transaksi pemasukan bahan makanan ke dalam gudang atau tempat penyimpanan.</p>
<p><b>Pre-kondisi</b></p>	<ul style="list-style-type: none"> <li>- Pegawai Katering telah login ke dalam sistem.</li> <li>- Data bahan yang akan dicatat sudah ada di sistem (dibuat oleh Admin).</li> </ul>

<b>Skenario Utama (Alur Sukses)</b>	<ol style="list-style-type: none"> <li>1. Pegawai memilih menu "Catat Stok Masuk".</li> <li>2. Sistem menampilkan formulir pencatatan stok masuk.</li> <li>3. Pegawai memilih bahan makanan dari daftar atau melalui fitur pencarian.</li> <li>4. Pegawai memasukkan jumlah stok yang masuk dan tanggal penerimaan.</li> <li>5. Pegawai menekan tombol "Simpan".</li> <li>6. Sistem memvalidasi input, lalu menambahkan jumlah stok pada data bahan terkait.</li> <li>7. <b>(Include) Sistem menjalankan use case "Periksa Stok Minimum"</b> untuk memperbarui status visual bahan tersebut (misalnya dari "Stok Menipis" menjadi "Normal").</li> <li>8. Sistem menyimpan catatan transaksi dan menampilkan pesan "Transaksi stok masuk berhasil dicatat".</li> </ol>
<b>Skenario Alternatif</b>	<p><b>A. Data Input Tidak Valid:</b></p> <p>6a. Jika jumlah yang dimasukkan bukan angka positif, sistem akan menampilkan pesan kesalahan dan meminta Pegawai untuk memperbaiki input.</p>
<b>Post-kondisi</b>	<ul style="list-style-type: none"> <li>- Jumlah stok untuk bahan yang dipilih telah bertambah.</li> <li>- Sebuah catatan transaksi stok masuk baru telah dibuat di dalam sistem.</li> </ul>

#### 3.4.1.5 Use Case: Catat Stok Keluar

<b>Nama Use Case</b>	Catat Stok Keluar
<b>Aktor</b>	Pegawai Katering

<b>Deskripsi Singkat</b>	Pegawai mencatat transaksi penggunaan atau pengeluaran bahan makanan untuk keperluan produksi.
<b>Pre-kondisi</b>	<ul style="list-style-type: none"> <li>- Pegawai Katering telah login ke dalam sistem.</li> <li>- Bahan yang akan dikeluarkan memiliki stok yang cukup.</li> </ul>

<b>Skenario Utama (Alur Sukses)</b>	<ol style="list-style-type: none"> <li>1. Pegawai memilih menu "Catat Stok Keluar".</li> <li>2. Sistem menampilkan formulir pencatatan stok keluar.</li> <li>3. Pegawai memilih bahan makanan dari daftar.</li> <li>4. Pegawai memasukkan jumlah stok yang digunakan.</li> <li>5. Pegawai menekan tombol "Simpan".</li> <li>6. Sistem memvalidasi input, lalu mengurangi jumlah stok pada data bahan terkait.</li> <li>7. <b>(Include) Sistem menjalankan use case "Periksa Stok Minimum"</b> untuk mengecek apakah stok saat ini berada di bawah batas minimum. Jika ya, sistem akan memberikan notifikasi visual.</li> <li>8. Sistem menyimpan catatan transaksi dan menampilkan pesan "Transaksi stok keluar berhasil dicatat".</li> </ol>
<b>Skenario Alternatif</b>	<p><b>A. Stok Tidak Cukup:</b></p> <p>6a. Jika jumlah yang akan dikeluarkan lebih besar dari stok yang tersedia, sistem akan menampilkan pesan kesalahan "Stok tidak mencukupi" dan transaksi dibatalkan.</p>
<b>Post-kondisi</b>	<ul style="list-style-type: none"> <li>- Jumlah stok untuk bahan yang dipilih telah berkurang.</li> <li>- Sebuah catatan transaksi stok keluar baru telah dibuat di dalam sistem.</li> </ul>

#### 3.4.1.6 Use Case: Pantau Status Stok

<b>Nama Use Case</b>	Pantau Status Stok
<b>Aktor</b>	Admin (Pemilik), Pegawai Katering
<b>Deskripsi Singkat</b>	Pengguna (Admin atau Pegawai) melihat ringkasan atau dasbor yang menampilkan status stok terkini dari semua bahan makanan.
<b>Pre-kondisi</b>	Pengguna telah login ke dalam sistem.
<b>Skenario Utama (Alur Sukses)</b>	<ol style="list-style-type: none"> <li>1. Pengguna mengakses halaman utama atau menu "Dasbor/Pantau Stok".</li> <li>2. Sistem mengambil data terkini semua bahan makanan.</li> <li>3. Sistem menampilkan daftar bahan beserta jumlah stok saat ini dan indikator status (misalnya: warna hijau untuk "Normal", kuning</li> </ol>

	untuk "Stok Menipis", merah untuk "Habis").
<b>Skenario Tambahan (Extend)</b>	<b>A. Pengecekan Stok Minimum:</b> Selama pengguna melihat halaman ini (langkah 3), <b>(Extend) use case "Periksa Stok Minimum" dapat dijalankan</b> untuk memperbarui indikator status secara dinamis, misalnya saat halaman di- <i>refresh</i> atau ada pembaruan data dari transaksi lain. Ini memastikan status yang ditampilkan selalu akurat.
<b>Post-kondisi</b>	Pengguna mendapatkan informasi visual mengenai kondisi stok bahan makanan secara <i>real-time</i> .

#### 3.4.1.7 Use Case: Lihat Laporan Stok

<b>Nama Use Case</b>	Lihat Laporan Stok
<b>Aktor</b>	Admin (Pemilik)
<b>Deskripsi Singkat</b>	Admin melihat dan mengunduh laporan historis transaksi stok dalam periode tertentu.
<b>Pre-kondisi</b>	Admin telah login ke dalam sistem.
<b>Skenario Utama (Alur Sukses)</b>	<ol style="list-style-type: none"> <li>1. Admin memilih menu "Laporan Stok".</li> <li>2. Sistem menampilkan opsi filter untuk laporan (misalnya: rentang tanggal, jenis transaksi, nama bahan).</li> <li>3. Admin memilih filter yang diinginkan dan menekan "Tampilkan Laporan".</li> <li>4. Sistem menampilkan laporan dalam bentuk tabel di layar.</li> <li>5. Admin menekan tombol "Ekspor ke Excel/CSV".</li> <li>6. Sistem membuat file laporan dan memulai proses unduhan di peramban Admin.</li> </ol>
<b>Skenario Alternatif</b>	<b>A. Data Tidak Ditemukan:</b> 4a. Jika tidak ada transaksi yang sesuai dengan filter yang dipilih, sistem akan menampilkan pesan "Data tidak



	ditemukan".
<b>Post-kondisi</b>	<ul style="list-style-type: none"> <li>- Admin telah melihat data laporan stok.</li> <li>- File laporan telah berhasil diunduh (jika opsi ekspor dipilih).</li> </ul>

## 4. Lampiran

### 4.1 Dukungan dan Pemeliharaan

- **Garansi:** Vendor akan memberikan dukungan teknis selama 3 bulan sejak sistem mulai digunakan.
- **Perbaikan Bug:** Perbaikan untuk *bug* atau cacat kecil akan diberikan secara gratis selama masa garansi.
- **Manajemen Perubahan:** Permintaan perubahan atau penambahan fitur besar di luar ruang lingkup awal dapat dinegosiasikan melalui kontrak terpisah.
- **Pelatihan:** Vendor wajib memberikan sesi pelatihan secara langsung kepada pemilik usaha atau satu staf kunci yang ditunjuk, dengan metode yang sederhana dan mudah dipahami.
- **Dokumentasi:** Semua dokumentasi terkait sistem (termasuk panduan pengguna) akan diserahkan pada akhir proyek.