**SKPL**-0104

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

Sistem Pembukuan

untuk:

Bakso Najwa

Dipersiapkan oleh:

Crisnandra Rahmita M. 1301160103

Mochammad Aria P.S. 1301160200

Gary Andersen 1301164540

Dinda Mareta Putriany 1301164558

Program web Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |
| --- | --- | --- | --- |
| Program web Studi S1 Teknik In*form*atika  -  Fakultas In*form*atika | Nomor Dokumen | | Halaman |
| *SKPL-0104* | | 45 |
| Revisi | *D* | *Tgl: 14 Mei 2018* |

# Daftar Perubahan

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A | Perbaikan nomor SKPL di footer  Perbaikan konvensi dokumen  Perbaikan referensi dan fungsi produk  Perbaikan karakteristik pengguna  Perbaikan lingkungan operasi  Perbaikan batasan perancangan dan implementasi  Perbaikan dokumentasi pengguna  Perbaikan use case scenario  Perbaikan antarmuka eksternal  Perbaikan lampiran B |
| B | Perbaikan nomor SKPL di footer  Perbaikan konvensi dokumen  Perbaikan karakteristik pengguna  Perbaikan lingkungan operasi  Perbaikan batasan perancangan dan implementasi  Perbaikan antarmuka eksternal  Penambahan model domain  Perbaikan lampiran B |
| C | Perbaikan tabel karakteristik pengguna  Perbaikan entity relationship diagram  Perbaikan class diagram |
| D | Penambahan Use case diagram  Penambahan Use case scenario  Penambahan atribut pada ERD |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX | - | A | B | C | D | E | F | G |
| TGL | 5 MARET 2018 | 2 APRIL 2018 | 15 APRIL 2018 | 30 APRIL 2018 | 14 MEI 2018 |  |  |  |
| Ditulis oleh | KELOMPOK 4 | KELOMPOK 4 | KELOMPOK 4 | KELOMPOK 4 | KELOMPOK 4 |  |  |  |
| Diperiksa oleh | IBU VERO | IBU VERO | IBU VERO | IBU VERO | IBU VERO |  |  |  |
| Disetujui oleh | IBU VERO | IBU VERO | IBU VERO | IBU VERO | IBU VERO |  |  |  |

# Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
| 16  36-37  15 | Penambahan Use case diagram  Penambahan Use case scenario  Penambahan entitas pada ERD |  |  |

# Daftar Isi

[Daftar Perubahan 1](#_Toc533037790)

[Daftar Halaman Perubahan 2](#_Toc533037791)

[Daftar Isi 3](#_Toc533037792)

[1. Pendahuluan 9](#_Toc533037793)

[1.1 Tujuan Penulisan Dokumen 9](#_Toc533037794)

[1.2 Konvensi Dokumen 9](#_Toc533037795)

[1.3 Cakupan Produk 10](#_Toc533037796)

[1.4 Referensi 10](#_Toc533037797)

[2. Overall Description 10](#_Toc533037798)

[2.1 Perspektif Produk 10](#_Toc533037799)

[2.2 Fungsi Produk 11](#_Toc533037800)

[2.3 Kelas dan Karakteristik Pengguna 11](#_Toc533037801)

[2.4 Lingkungan Operasi 12](#_Toc533037802)

[2.5 Batasan Perancangan dan Implementasi 12](#_Toc533037803)

[2.6 Dokumentasi Pengguna 12](#_Toc533037804)

[2.7 Asumsi dan Dependensi 13](#_Toc533037805)

[3. Requirements Antarmuka Eksternal 13](#_Toc533037806)

[3.1 Antarmuka Pengguna 13](#_Toc533037807)

[3.2 Antarmuka Perangkat Keras 14](#_Toc533037808)

[3.3 Antarmuka Perangkat Lunak 14](#_Toc533037809)

[3.4 Antarmuka Komunikasi 14](#_Toc533037810)

[4. Model Domain 15](#_Toc533037811)

[4.1 Entity Relationship Diagram 15](#_Toc533037812)

[4.2 Class Diagram 16](#_Toc533037813)

[5. Fitur Sistem (Use Cases) 17](#_Toc533037814)

[5.1 Use Case *Login* 18](#_Toc533037815)

[5.1.1. Nama Use Case: Login 18](#_Toc533037816)

[5.1.2. Tujuan: Agar aktor (Staff Gudang, Service, Keuangan, Pemilik) dapat menggunakan program web dengan otoritasnya masing masing 18](#_Toc533037817)

[5.1.3. Input: username dan password 18](#_Toc533037818)

[5.1.4. Output: data bahan, data produk, data pemesanan, data pengeluaran, dan data karyawan 18](#_Toc533037819)

[5.1.5. Skenario Utama: Aktor memiliki hak login dan harus melakukan login agar dapat menggunakan program web 18](#_Toc533037820)

[5.1.6. Prakondisi: Aktor ingin menggunakan program web dan belum melakukan login. 18](#_Toc533037821)

[5.1.7. Langkah-langkah: 18](#_Toc533037822)

[5.1.8. Pascakondisi: Aktor telah melakukan login dan siap berinteraksi dengan program web 18](#_Toc533037823)

[5.1.9. Contoh: Pemilik ingin menggunakan program web yang ada, sebelum melakukan itu pemilik harus memasukkan username dan password untuk login agar dapat masuk ke dalam halaman utama program web. 18](#_Toc533037824)

[5.2 Use Case *Input* Bahan 18](#_Toc533037825)

[5.2.1. Nama Use Case: Input Bahan 18](#_Toc533037826)

[5.2.2. Tujuan: Untuk mencatat, menambahkan, dan menyimpan data bahan yang baru 18](#_Toc533037827)

[5.2.3. Input: Data bahan (idbahan, nama bahan, jumlah stok bahan, tanggal) 18](#_Toc533037828)

[5.2.4. Output: Data bahan (idbahan, nama bahan, jumlah stok bahan, tanggal) yang tersimpan dalam database bahan 18](#_Toc533037829)

[5.2.5. Skenario Utama: Staff Gudang mempunyai hak input bahan untuk menginput data bahan 18](#_Toc533037830)

[5.2.6. Prakondisi: Jika ingin menggunakan input bahan, Staff Gudang telah login untuk menggunakan menu input bahan dan data bahan belum tersimpan di database bahan 18](#_Toc533037831)

[5.2.7. Langkah-langkah: 18](#_Toc533037832)

[5.2.8. Pascakondisi: Staff Gudang telah menginputkan data bahan dan sistem berhasil menyimpan data bahan di database bahan 19](#_Toc533037833)

[5.2.9. Contoh: Staff Gudang ingin input data bahan tepung. Staff Gudang yang merupakan Staff Gudang melakukan input data bahan melalui menu input bahan. Sistem mencatat dan menyimpan kedalam database bahan 19](#_Toc533037834)

[5.3 Use Case *View* Bahan 19](#_Toc533037835)

[5.3.1 Nama Use Case: View Bahan 19](#_Toc533037836)

[5.3.2 Tujuan: Untuk melihat data bahan 19](#_Toc533037837)

[5.3.3 Input: idbahan/nama bahan 19](#_Toc533037838)

[5.3.4 Output: Data bahan (idbahan, nama bahan, jumlah stok bahan,tanggal) 19](#_Toc533037839)

[5.3.5 Skenario Utama: Staff Gudang mempunyai hak view bahan untuk melihat data bahan yang tersedia 19](#_Toc533037840)

[5.3.6 Prakondisi: Jika ingin menggunakan view bahan, Staff Gudang telah login untuk menggunakan menu view bahan dan data bahan tersedia di database bahan 19](#_Toc533037841)

[5.3.7 Langkah-langkah: 19](#_Toc533037842)

[5.3.8 Pascakondisi: Staff Gudang dapat melihat data bahan di database bahan 20](#_Toc533037843)

[5.3.9 Contoh: Pengguna ingin melihat data bahan tepung. Pengguna yang merupakan Staff Gudang mengecek bahan melalui menu view bahan dan sistem menampilkan data bahan 20](#_Toc533037844)

[5.4 Use Case *Edit* Bahan 20](#_Toc533037845)

[5.4.1 Nama Use Case: Edit Bahan 20](#_Toc533037846)

[5.4.2 Tujuan: Memperbaharui data bahan 20](#_Toc533037847)

[5.4.3 Input: Data bahan (idbahan/nama bahan, jumlah stok bahan, tanggal) yang terbaru 20](#_Toc533037848)

[5.4.4 Output: Data bahan terbaru yang tersimpan di dalam database bahan 20](#_Toc533037849)

[5.4.5 Skenario Utama: Staff Gudang mempunyai hak edit bahan untuk memperbaharui data bahan yang tersedia 20](#_Toc533037850)

[5.4.6 Prakondisi: Jika ingin menggunakan edit bahan, Staff Gudang telah login untuk menggunakan menu edit bahan dan data bahan tersedia di database bahan 20](#_Toc533037851)

[5.4.7 Langkah-langkah: 20](#_Toc533037852)

[5.4.8 Pascakondisi: Staff Gudang telah meng-edit data bahan dan sistem berhasil memperbaharui data bahan di database bahan 21](#_Toc533037853)

[5.4.9 Contoh: Pengguna ingin memperbaharui data bahan tepung. Pengguna yang merupakan Staff Gudang mengecek bahan melalui menu edit bahan dan sistem menampilkan data bahan. Setelah menemukan dan merubah data bahan, sistem memperbaharui data. 21](#_Toc533037854)

[5.5 Use Case *Delete* Bahan 21](#_Toc533037855)

[5.5.1 Nama Use Case: Delete Bahan 21](#_Toc533037856)

[5.5.2 Tujuan: Menghapus data bahan 21](#_Toc533037857)

[5.5.3 Input: Id bahan/nama bahan yang akan di delete 21](#_Toc533037858)

[5.5.4 Output: data bahan pada database bahan terbaru 21](#_Toc533037859)

[5.5.5 Skenario Utama: Staff Gudang mempunyai hak delete bahan untuk menghapus data bahan pada database 21](#_Toc533037860)

[5.5.6 Prakondisi: Jika ingin menggunakan delete bahan, Staff Gudang telah login untuk menggunakan menu delete data bahan tersedia di database bahan 21](#_Toc533037861)

[5.5.7 Langkah-langkah: 21](#_Toc533037862)

[5.5.8 Pascakondisi: Staff Gudang telah menghapus data bahan dan sistem berhasil memperbaharui data bahan di database bahan 22](#_Toc533037863)

[5.5.9 Contoh: Staff Gudang menghapus data bahan tepung dari database 22](#_Toc533037864)

[5.6 Use Case *Input* Produk 22](#_Toc533037865)

[5.6.1 Nama Use Case: Input Produk 22](#_Toc533037866)

[5.6.2 Tujuan: Untuk mencatat, menambahkan, dan menyimpan data produk yang baru 22](#_Toc533037867)

[5.6.3 Input: Data produk (id produk, nama produk, tanggal, tipe produk, harga produk) 22](#_Toc533037868)

[5.6.4 Output: Data produk yang tersimpan dalam database produk 22](#_Toc533037869)

[5.6.5 Skenario Utama: Staff Gudang mempunyai hak input bahan untuk menginput data produk 22](#_Toc533037870)

[5.6.6 Prakondisi: Jika ingin menggunakan input produk, Staff Gudang telah login untuk menggunakan menu input produk dan data produk belum tersimpan di database produk 22](#_Toc533037871)

[5.6.7 Langkah-langkah: 22](#_Toc533037872)

[5.6.8 Pascakondisi: Staff Gudang telah menginputkan data produk dan sistem berhasil menyimpan data produk di database produk 22](#_Toc533037873)

[5.6.9 Contoh: Pengguna ingin input data produk bakso. Pengguna yang merupakan Staff Gudang melakukan input data produk melalui menu input produk. Sistem mencatat dan menyimpan kedalam database produk 23](#_Toc533037874)

[5.7 Use Case *View* Produk 23](#_Toc533037875)

[5.7.1 Nama Use Case: View Produk 23](#_Toc533037876)

[5.7.2 Tujuan: Untuk melihat data bahan 23](#_Toc533037877)

[5.7.3 Input: id produk/nama produk 23](#_Toc533037878)

[5.7.4 Output: Data produk (id produk, nama produk, tanggal, tipe produk, harga produk) yang tersimpan dalam database produk 23](#_Toc533037879)

[5.7.5 Skenario Utama: Staff Gudang atau service mempunyai hak view produk untuk melihat data produk yang tersedia 23](#_Toc533037880)

[5.7.6 Prakondisi: Jika ingin menggunakan view produk, Staff Gudang atau service telah login untuk menggunakan menu view produk dan data produk tersedia di database produk 23](#_Toc533037881)

[5.7.7 Langkah-langkah: 23](#_Toc533037882)

[5.7.8 Pascakondisi: Staff Gudang atau service dapat melihat data produk di database produk 23](#_Toc533037883)

[5.7.9 Contoh: Pengguna ingin melihat data produk bakso. Pengguna yang merupakan Staff Gudang atau service mengecek produk melalui menu view produk dan sistem menampilkan data produk 23](#_Toc533037884)

[5.8 Use Case *Edit* Produk 23](#_Toc533037885)

[5.8.1 Nama Use Case: Edit Produk 23](#_Toc533037886)

[5.8.2 Tujuan: Memperbaharui data produk 23](#_Toc533037887)

[5.8.3 Input: Data produk (id produk, jumlah stok produk, nama produk, tanggal, tipe produk, harga produk) 23](#_Toc533037888)

[5.8.4 Output: Data produk terbaru yang tersimpan di dalam database produk 23](#_Toc533037889)

[5.8.5 Skenario Utama: Staff Gudang atau service mempunyai hak edit produk untuk memperbaharui data produk yang tersedia 23](#_Toc533037890)

[5.8.6 Prakondisi: Jika ingin menggunakan edit produk, Staff Gudang atau service telah login untuk menggunakan menu edit produk dan data produk tersedia di database produk 23](#_Toc533037891)

[5.8.7 Langkah-langkah: 24](#_Toc533037892)

[5.8.8 Pascakondisi: Staff Gudang atau service telah meng-edit data produk dan sistem berhasil memperbaharui data produk di database produk 24](#_Toc533037893)

[5.8.9 Contoh: Pengguna ingin memperbaharui data produk bakso. Pengguna yang merupakan Staff Gudang atau service mengecek produk melalui menu edit produk dan sistem menampilkan data produk Setelah menemukan dan merubah data produk, sistem memperbaharui data. 24](#_Toc533037894)

[5.9 Use Case *Delete* Produk 24](#_Toc533037895)

[5.9.1 Nama Use Case: Delete Produk 24](#_Toc533037896)

[5.9.2 Tujuan: Menghapus data produk 24](#_Toc533037897)

[5.9.3 Input: data produk (idproduk/nama produk) yang akan di delete 24](#_Toc533037898)

[5.9.4 Output: data produk pada database produk terbaru 24](#_Toc533037899)

[5.9.5 Skenario Utama: Staff Gudang mempunyai hak delete bahan untuk menghapus data produk pada database 24](#_Toc533037900)

[5.9.6 Prakondisi: Jika ingin menggunakan delete produk, Staff Gudang telah login untuk menggunakan menu delete data produk tersedia di database produk 24](#_Toc533037901)

[5.9.7 Langkah-langkah: 24](#_Toc533037902)

[5.9.8 Pascakondisi: Staff Gudang telah menghapus data produk dan sistem berhasil memperbaharui data bahan di database produk 25](#_Toc533037903)

[5.9.9 Contoh: Pengguna menghapus data produk bakso dari database 25](#_Toc533037904)

[5.10 Use Case *Input* Pemesanan 25](#_Toc533037905)

[5.10.1 Nama Use Case: input pemesanan 25](#_Toc533037906)

[5.10.2 Tujuan: Untuk mencatat, menambahkan, dan menyimpan data pemesanan yang baru 25](#_Toc533037907)

[5.10.3 Input: data pemesanan (nama konsumen, nama instansi, kontak personal, alamat, pesanan, total harga pemesanan, status pembayaran, tanggal pemesanan, tanggal pembayaran, id pemesanan) 25](#_Toc533037908)

[5.10.4 Output: data pemesanan yang tersimpan dalam database pemesanan 25](#_Toc533037909)

[5.10.5 Skenario Utama: Service mempunyai hak input pemesanan untuk menginput data pemesanan 25](#_Toc533037910)

[5.10.6 Prakondisi: Jika ingin menggunakan input pemesanan, Service telah login untuk menggunakan menu input pemesanan dan data pemesanan belum tersimpan di database pemesanan 25](#_Toc533037911)

[5.10.7 Langkah-langkah: 25](#_Toc533037912)

[5.10.8 Pascakondisi: Service telah menginputkan data pemesanan dan sistem berhasil menyimpan data pemesanan di database pemesanan 26](#_Toc533037913)

[5.10.9 Contoh: Pengguna ingin input data pemesanan bakso. Pengguna yang merupakan Service melakukan input data pemesanan melalui menu input pemesanan. Sistem mencatat dan menyimpan kedalam database pemesanan 26](#_Toc533037914)

[5.11 Use Case *View* Pemesanan 26](#_Toc533037915)

[5.11.1 Nama Use Case: view pemesanan 26](#_Toc533037916)

[5.11.2 Tujuan: Untuk melihat data pemesanan 26](#_Toc533037917)

[5.11.3 Input: idPemesanan/nama konsumen/tanggal pemesanan 26](#_Toc533037918)

[5.11.4 Output: data pemesanan 26](#_Toc533037919)

[5.11.5 Skenario Utama: Service, Staff Gudang, Keuangan atau Pemilik mempunyai hak view pemesanan untuk melihat data pemesanan yang tersedia 26](#_Toc533037920)

[5.11.6 Prakondisi: Jika ingin menggunakan view pemesanan, Service, Staff Gudang, Keuangan atau Pemilik telah login untuk menggunakan menu view pemesanan dan data pemesanan tersedia di database pemesanan 26](#_Toc533037921)

[5.11.7 Langkah-langkah: 26](#_Toc533037922)

[5.11.8 Pascakondisi: Service, Staff Gudang, Keuangan atau Pemilik telah melihat data pemesanan 27](#_Toc533037923)

[5.11.9 Contoh: Pengguna ingin melihat data pemesanan. Pengguna yang merupakan Service, Staff Gudang, Keuangan atau Pemilik mengecek data pemesanan melalui menu view pemesanan dan sistem menampilkan data pemesanan 27](#_Toc533037924)

[5.12 Use Case *Edit* Pemesanan 27](#_Toc533037925)

[5.12.1 Nama Use Case: edit pemesanan 27](#_Toc533037926)

[5.12.2 Tujuan: Memperbaharui data pemesanan 27](#_Toc533037927)

[5.12.3 Input: data pemesanan (nama konsumen, nama instansi, kontak personal, alamat, pesanan, total harga pemesanan, status pembayaran, tanggal pemesanan, tanggal pembayaran,id pemesanan) 27](#_Toc533037928)

[5.12.4 Output: data pemesanan terbaru yang tersimpan di dalam database pemesanan 27](#_Toc533037929)

[5.12.5 Skenario Utama: Service atau Keuangan mempunyai hak edit pemesanan untuk memperbaharui data pemesanan yang tersedia 27](#_Toc533037930)

[5.12.6 Prakondisi: Jika ingin menggunakan edit pemesanan, Service atau Keuangan telah login untuk menggunakan menu edit pemesanan dan data pemesanan tersedia di database pemesanan 27](#_Toc533037931)

[5.12.7 Langkah-langkah: 27](#_Toc533037932)

[5.12.8 Pascakondisi: Service atau Keuangan telah meng-edit data pemesanan dan sistem berhasil memperbaharui data pemesanan di database pemesanan 28](#_Toc533037933)

[5.12.9 Contoh: Pengguna ingin memperbaharui data pemesanan bakso. Pengguna yang merupakan Service dan Keuangan mengecek pemesanan melalui menu edit pemesanan dan sistem menampilkan data pemesanan. Setelah menemukan dan merubah data pemesanan, sistem memperbaharui data. 28](#_Toc533037934)

[5.13 Use Case *Delete* Pemesanan 28](#_Toc533037935)

[5.13.1 Nama Use Case: delete pemesanan 28](#_Toc533037936)

[5.13.2 Tujuan: Menghapus data pemesanan 28](#_Toc533037937)

[5.13.3 Input: data pemesanan (id pemesanan/nama konsumen) yang akan di delete 28](#_Toc533037938)

[5.13.4 Output: data pemesanan terbaru 28](#_Toc533037939)

[5.13.5 Skenario Utama: Service mempunyai hak delete pemesanan untuk menghapus data pemesanan pada database 28](#_Toc533037940)

[5.13.6 Prakondisi: Jika ingin menggunakan delete pemesanan, Service telah login untuk menggunakan menu delete pemesanan dan data pemesanan tersedia di database pemesanan. 28](#_Toc533037941)

[5.13.7 Langkah-langkah: 28](#_Toc533037942)

[5.13.8 Pascakondisi: Service telah menghapus data pemesanan dan sistem berhasil memperbaharui data pemesanan di database pemesanan 29](#_Toc533037943)

[5.13.9 Contoh: Service menghapus data pemesanan bakso dari database pemesanan 29](#_Toc533037944)

[5.14 Use Case *Input* Pengeluaran 29](#_Toc533037945)

[5.14.1 Nama Use Case: Input Pengeluaran 29](#_Toc533037946)

[5.14.2 Tujuan: Untuk mencatat, menambahkan, dan menyimpan data pengeluaran yang baru 29](#_Toc533037947)

[5.14.3 Output: Data pengeluaran yang tersimpan dalam database pengeluaran 29](#_Toc533037948)

[5.14.4 Input: data pengeluaran (keterangan, harga pengeluaran, tanggal pengeluaran) 29](#_Toc533037949)

[5.14.5 Skenario Utama: Keuangan mempunyai hak input pengeluaran untuk menginput data pengeluaran 29](#_Toc533037950)

[5.14.6 Prakondisi: Jika ingin menggunakan input pengeluaran, Keuangan telah login untuk menggunakan menu input pengeluaran dan data pengeluaran belum tersimpan di database pengeluaran 29](#_Toc533037951)

[5.14.7 Langkah-langkah: 29](#_Toc533037952)

[5.14.8 Pascakondisi: Keuangan telah menginputkan data pengeluaran dan sistem berhasil menyimpan data pengeluaran di database pengeluaran 29](#_Toc533037953)

[5.14.9 Contoh: Pengguna ingin input data pengeluaran listrik. Pengguna yang merupakan Keuangan melakukan input data pengeluaran melalui menu input pengeluaran. Sistem mencatat dan menyimpan kedalam database pengeluaran 30](#_Toc533037954)

[5.15 Use Case *View* Pengeluaran 30](#_Toc533037955)

[5.15.1 Nama Use Case: View Pengeluaran 30](#_Toc533037956)

[5.15.2 Tujuan: Untuk melihat data pengeluaran 30](#_Toc533037957)

[5.15.3 Input: idPengeluaran/ keterangan/ tanggal pengeluaran 30](#_Toc533037958)

[5.15.4 Output: Untuk melihat data pengeluaran 30](#_Toc533037959)

[5.15.5 Skenario Utama: Keuangan atau Pemilik mempunyai hak view pengeluaran untuk melihat data pengeluaran yang tersedia 30](#_Toc533037960)

[5.15.6 Prakondisi: Jika ingin menggunakan view pengeluaran, Keuangan atau Pemilik telah login untuk menggunakan menu view pengeluaran dan data pengeluaran tersedia di database pengeluaran 30](#_Toc533037961)

[5.15.7 Langkah-langkah: 30](#_Toc533037962)

[5.15.8 Pascakondisi: Keuangan atau Pemilik telah melihat data pengeluaran 30](#_Toc533037963)

[5.15.9 Contoh: Pengguna ingin melihat data pengeluaran. Pengguna yang merupakan Keuangan atau Pemilik mengecek data pengeluaran apa saja melalui menu view pengeluaran dan sistem menampilkan data pengeluaran 30](#_Toc533037964)

[5.16 Use Case *Edit* Pengeluaran 30](#_Toc533037965)

[5.16.1 Nama Use Case: Edit Pengeluaran 30](#_Toc533037966)

[5.16.2 Tujuan: Memperbaharui data pengeluaran 30](#_Toc533037967)

[5.16.3 Input: Data pengeluaran (id pengeluaran, keterangan, harga pengeluaran, tanggal pengeluaran) yang terbaru 30](#_Toc533037968)

[5.16.4 Output: Data pengeluaran terbaru yang tersimpan di dalam database pengeluaran 30](#_Toc533037969)

[5.16.5 Skenario Utama: Keuangan mempunyai hak edit pengeluaran untuk memperbaharui data pengeluaran yang tersedia 30](#_Toc533037970)

[5.16.6 Prakondisi: Jika ingin menggunakan edit pengeluaran, Keuangan telah login untuk menggunakan menu edit pengeluaran dan data pengeluaran tersedia di database pengeluaran 31](#_Toc533037971)

[5.16.7 Langkah-langkah: 31](#_Toc533037972)

[5.16.8 Pascakondisi: Keuangan telah meng-edit data pengeluaran dan sistem berhasil memperbaharui data pengeluaran di database pengeluaran 31](#_Toc533037973)

[5.16.9 Contoh: Pengguna ingin memperbaharui data pengeluaran listrik. Pengguna yang merupakan Keuangan mengecek pengeluaran melalui menu edit pengeluaran dan sistem menampilkan data pengeluaran. Setelah menemukan dan merubah data pengeluaran, sistem memperbaharui data. 31](#_Toc533037974)

[5.17 Use Case *Delete* Pengeluaran 31](#_Toc533037975)

[5.17.1 Nama Use Case: Delete Pengeluaran 31](#_Toc533037976)

[5.17.2 Tujuan: Menghapus data pengeluaran 31](#_Toc533037977)

[5.17.3 Input: Data pengeluaran (keterangan) yang akan di delete 31](#_Toc533037978)

[5.17.4 Output: Data pengeluaran terbaru 31](#_Toc533037979)

[5.17.5 Skenario Utama: Keuangan mempunyai hak delete pengeluaran untuk menghapus data pengeluaran pada database 31](#_Toc533037980)

[5.17.6 Prakondisi: Jika ingin menggunakan delete pengeluaran, Keuangan telah login untuk menggunakan menu delete pengeluaran dan data pengeluaran tersedia di database tambahan 32](#_Toc533037981)

[5.17.7 Langkah-langkah: 32](#_Toc533037982)

[5.17.8 Pascakondisi: Keuangan telah menghapus data pengeluaran dan sistem berhasil memperbaharui data pengeluaran di database pengeluaran 32](#_Toc533037983)

[5.17.9 Contoh: Keuangan menghapus data pengeluaran dari database 32](#_Toc533037984)

[5.18 Use Case *Input* Karyawan 32](#_Toc533037985)

[5.18.1 Nama Use Case: Input Karyawan 32](#_Toc533037986)

[5.18.2 Tujuan: Untuk menambahkan data karyawan yang baru 32](#_Toc533037987)

[5.18.3 Input: data karyawan (Nama karyawan, gaji, NIK, divisi, nomor HP) 32](#_Toc533037988)

[5.18.4 Output: Data karyawan yang tersimpan dalam database karyawan 32](#_Toc533037989)

[5.18.5 Skenario Utama: Pemilik mempunyai hak input karyawan untuk menginput data karyawan 32](#_Toc533037990)

[5.18.6 Prakondisi: Jika ingin menggunakan input karyawan, Pemilik telah login untuk menggunakan menu input karyawan dan data karyawan belum tersimpan di database karyawan 32](#_Toc533037991)

[5.18.7 Langkah-langkah: 32](#_Toc533037992)

[5.18.8 Pascakondisi: Pemilik telah menginputkan data karyawan dan sistem berhasil menyimpan data karyawan di database karyawan 33](#_Toc533037993)

[5.18.9 Contoh: Pengguna ingin input data karyawan baru. Pengguna yang merupakan Pemilik melakukan input data karyawan melalui menu input karyawan. Sistem mencatat dan menyimpan kedalam database karyawan 33](#_Toc533037994)

[5.19 Use Case *View* Karyawan 33](#_Toc533037995)

[5.19.1 Nama Use Case: View Karyawan 33](#_Toc533037996)

[5.19.2 Tujuan: Untuk melihat data karyawan 33](#_Toc533037997)

[5.19.3 Input: data karyawan (Nama Karyawan/NIK) 33](#_Toc533037998)

[5.19.4 Output: Data Karyawan 33](#_Toc533037999)

[5.19.5 Skenario Utama: Pemilik atau keuangan mempunyai hak view karyawan untuk melihat data biaya karyawan yang tersedia 33](#_Toc533038000)

[5.19.6 Prakondisi: Jika ingin menggunakan view karyawan, Pemilik atau keuangan telah login untuk menggunakan menu view data karyawan dan data karyawan tersedia di database karyawan 33](#_Toc533038001)

[5.19.7 Langkah-langkah: 33](#_Toc533038002)

[5.19.8 Pascakondisi: Pemilik atau keuangan telah melihat data karyawan 34](#_Toc533038003)

[5.19.9 Contoh: Pemilik atau keuangan ingin melihat data karyawan. Pemilik atau keuangan mengecek karyawan siapa saja melalui tabel view karyawan dan sistem menampilkan data karyawan 34](#_Toc533038004)

[5.20 Use Case *Edit* Karyawan 34](#_Toc533038005)

[5.20.1 Nama Use Case: Edit Karyawan 34](#_Toc533038006)

[5.20.2 Tujuan: Memperbaharui data karyawan 34](#_Toc533038007)

[5.20.3 Input: Data karyawan yang terbaru (Nama karyawan, gaji, NIK, divisi, nomor HP) 34](#_Toc533038008)

[5.20.4 Output: Data karyawan terbaru yang tersimpan di dalam database karyawan 34](#_Toc533038009)

[5.20.5 Skenario Utama: Pemilik mempunyai hak edit karyawan untuk memperbaharui data karyawan yang tersedia 34](#_Toc533038010)

[5.20.6 Prakondisi: Jika ingin menggunakan edit karyawan, Pemilik telah login untuk menggunakan menu edit karyawan dan data karyawan tersedia di database karyawan 34](#_Toc533038011)

[5.20.7 Langkah-langkah: 34](#_Toc533038012)

[5.20.8 Pascakondisi: Pemilik telah meng-edit data karyawan dan sistem berhasil memperbaharui data karyawan di database karyawan 35](#_Toc533038013)

[5.20.9 Contoh: Pengguna ingin memperbaharui data karyawan X. Pengguna yang merupakan Pemilik mengecek karyawan melalui menu edit karyawan dan sistem menampilkan data karyawan. Setelah menemukan dan merubah data karyawan, sistem memperbaharui data. 35](#_Toc533038014)

[5.21 Use Case *Delete* Karyawan 35](#_Toc533038015)

[5.21.1 Nama Use Case: Delete Karyawan 35](#_Toc533038016)

[5.21.2 Tujuan: Menghapus data karyawan 35](#_Toc533038017)

[5.21.3 Input: Data karyawan yang akan di delete (nama karyawan/NIK) 35](#_Toc533038018)

[5.21.4 Output: Data karyawan terbaru 35](#_Toc533038019)

[5.21.5 Skenario Utama: Pemilik mempunyai hak delete karyawan untuk menghapus data karyawan pada database 35](#_Toc533038020)

[5.21.6 Prakondisi: Jika ingin menggunakan delete karyawan, Pemilik telah login untuk menggunakan menu delete karyawan dan data karyawan tersedia di database karyawan 35](#_Toc533038021)

[5.21.7 Langkah-langkah: 35](#_Toc533038022)

[5.21.8 Pascakondisi: Pemilik telah menghapus data karyawan dan sistem berhasil memperbaharui data karyawan di database karyawan. 36](#_Toc533038023)

[5.21.9 Contoh: Pemilik menghapus data karyawan dari database 36](#_Toc533038024)

[6. Requirements Nonfungsional Lainnya 36](#_Toc533038025)

[6.1 Requirements Performa 36](#_Toc533038026)

[6.2 Requirements Keselamatan 36](#_Toc533038027)

[6.3 Requirements Keamanan 36](#_Toc533038028)

[6.4 Atribut Kualitas Perangkat Lunak 36](#_Toc533038029)

[6.4.1 Ketersediaan (Availability) 36](#_Toc533038030)

[6.4.2 Keandalan 36](#_Toc533038031)

[6.4.3 Kepeliharaan (maintainability) 36](#_Toc533038032)

[7. Requirements Lain 36](#_Toc533038033)

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini berisi Spesifikasi Kebutuhan Perangkat Lunak (SKPL) untuk perancangan Sistem Pembukuan pada Perusahaan Bakso Najwa. Tujuan dari penulisan dokumen ini adalah untuk memberikan penjelasan mengenai perangkat lunak yang akan dibangun baik berupa gambaran umum maupun penjelasan secara rinci kepada *developer* dan *client*.

Dokumen ini akan digunakan sebagai acuan dan sebagai bahan evaluasi proses pengembangan perangkat lunak pada tahap selanjutnya. Selain itu, dokumen ini digunakan agar perangkat lunak yang akan dibangun sesuai dengan kebutuhan yang diinginkan oleh *client*.

## Konvensi Dokumen

Dalam konvensi dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini terdapat beberapa *requirement* yang dibutuhkan. *Requirement* yang di **cetak tebal**, adalah *requirement* utama atau *requirement* prioritas. Berikut merupakan *requirement* yang dibutuhkan dalam proses perancangan perangkat lunak yang akan dibangun, diantaranya:

* **Menyimpan data pemesanan** (id pemesanan, nama konsumen, nama instansi, kontak personal, alamat, pesanan, status pembayaran, total harga pemesanan, tanggal pemesanan, tanggal pembayaran)
* **Menyimpan data bahan** (id bahan, nama bahan, jumlah stok bahan, tanggal)
* **Menyimpan data pengeluaran** (id pengeluaran, keterangan, harga pengeluaran, tanggal pengeluaran)
* **Menyimpan data produk** (id produk, nama produk, tanggal, tipe produk, harga produk)
* **Menyimpan data karyawan** (Nama karyawan, gaji, NIK, divisi, nomor HP)
* *Input* data pemesanan
* *Input* data bahan
* *Input* data produk
* *Input* data karyawan
* *Input* data pengeluaran
* *Edit* data pemesanan
* *Edit* data bahan
* *Edit* data produk
* *Edit* data karyawan
* *Edit* data pengeluaran
* *Delete* data pemesanan
* *Delete* data bahan
* *Delete* data produk
* *Delete* data karyawan
* *Delete* data pengeluaran
* *View* data bahan
* *View* data produk
* *View* data pemesanan
* *View* data karyawan
* *View* data pengeluaran
* Data pemesanan dikelola (*input*, *edit*, *delete*, *view*) oleh *Service* dan Keuangan dapat meng-*edit* data pemesanan
* Data bahan dan produk dikelola (*input*, *edit*, *delete*, *view*) oleh *Staff* Gudang
* Data pengeluaran dikelola (*input*, *edit*, *delete*, *view*) oleh Keuangan
* Data karyawan dikelola (*input*, *edit*, *delete*, *view*) oleh Pemilik
* Pemilik dapat melihat semua data yang ada di dalam system

## Cakupan Produk

Produk yang akan diusulkan adalah perangkat lunak untuk sistem pembukuan Bakso Najwa yang akan mencatatkan setiap data pengeluaran, data pemesanan, data barang, data produk, data bahan serta dokumentasi karyawan dan akan disimpan ke dalam *database* yang sudah dibuat. Sistem pembukuan ini berbasis web dan akan menampilkan rincian pembukuankepada *client* (pemilik), sehingga *client* dapat mengetahui laju perkembangan bisnisnya.

Sistem pembukuan ini akan dirancang untuk me*menu*hi kebutuhan *client*. Sistem Pembukuan Bakso Najwa dibangun dengan tujuan untuk:

* Menyimpan data secara terstruktur
* Mencatat data barang, produk, dan karyawan
* Mencatat data pemesanan dan pengeluaran

Adapun manfaat dari perancangan sistem pembukuan ini diantaranya:

* Sistem pembukuan ini akan memudahkan pemilik perusahaan untuk mengendalikan arus pengeluaran dan pemasukan keuangan
* Sistem pembukuan ini akan membantu mengelola segala proses pencatatanyang terjadi di dalam perusahaan ini agar lebih terstruktur dan jelas

## Referensi

Dokumen yang dijadikan panutan dan membantu untuk pembuatan dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini sebagai berikut:

* *Template* SKPL Analisis Berorientasi Objek Telkom University Fakultas In*form*atika
* GL01-802 Atau SKPL-802, Prodi S1-Teknik In*form*atika, Universitas Telkom, 2016

# Overall Description

## Perspektif Produk

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini produk yang akan dirancang merupakan produk yang disengaja ada untuk me*menu*hi kebutuhan *client*. *Client* belum pernah memiliki sistem tertentu sebelumnya. Oleh karena itu, produk yang akan dirancang merupakan *self-contained product* pertama yang akan dimiliki oleh *client*.

## Fungsi Produk

Permasalahan yang dimiliki oleh *client* antara lain pencacatan pembukuan perusahaan masih belum tercatat baik dan rapi sehingga membingungkan admin keuangan dan pemilik perusahaan (*client*). Belum adanya pembukuan pemesanan dan pengeluaran perusahaan yang tersimpan dengan baik, karena proses pembukuan ini masih manual yaitu dicatat dengan kertas. Ini memungkinan adanya nota pembelian yang hilang atau tercecer sehingga pembukuan pemesanan dan pengeluaran menjadi tidak sinkron. Pendataan data *customer* masih belum tercatat dengan baik, masih ada *customer* yang mengutang dan kadang *customer*-nya lupa untuk membayar barang yang telah ia ambil.

Oleh karena itu, dibutuhkan perangkat lunak yang dapat membantu *client* mengatasi masalahnya. Perangkat lunak ini digunakan untuk menyimpan dan mengelola data-data yang di perlukan dan digunakan oleh *client* secara terstruktur, seperti data bahan dan data produk dikelola oleh *staff* gudang, data pemesanan dikelola oleh *service* dan dapat di-*edit* atau dilihat keuangan, data karyawan dikelola oleh pemilik, dan data pengeluaran dikelola oleh bagian keuangan. Selain itu, perangkat lunak ini dapat menampilkan rincian pembukuan. Sehingga aplikasi sistem pembukuan ini memiliki beberapa fungsi utama, yaitu:

1. *Login*

Proses *login* dilakukan oleh setiap pengguna yang ingin menggunakan aplikasi. Saat melakukan *login* harus mengisi *username* dan *password*.

1. Kelola Pemesanan

Proses ini digunakan untuk mengelola, mencatat, dan menyimpan data pemesanan. Dalam proses ini, pengguna dapat melakukan proses *input*, *edit*, *delete*, dan *view*. Data Pemesaan berupa

1. Kelola Bahan

Proses ini digunakan untuk mengelola, mencatat, dan menyimpan data bahan. Dalam proses ini, pengguna dapat melakukan proses *input*, *edit*, *delete*, dan *view*.

1. Kelola Produk

Proses ini digunakan untuk mengelola, mencatat, dan menyimpan data produk. Dalam proses ini, pengguna dapat melakukan proses *input*, *edit*, *delete*, dan *view*.

1. Kelola Karyawan

Proses ini digunakan untuk mengelola, mencatat, dan menyimpan data karyawan. Dalam proses ini, pengguna dapat melakukan proses *input*, *edit*, *delete*, dan *view*.

1. Kelola Pengeluaran

Proses ini digunakan untuk mengelola, mencatat, dan menyimpan data pengeluaran. Dalam proses ini, pengguna dapat melakukan proses *input*, *edit, delete,* dan *view.*

1. Logout

Proses ini dilakukan oleh masing-masing pengguna untuk keluar dari sistem pembukuan.

## Kelas dan Karakteristik Pengguna

Pengguna perangkat lunak ini adalah admin di perusahaan dan *client* yang akan menerima perkembangan dan laporan-laporan pembukuan perusahaanya. Perangkat lunak ini hanya dibuat dan digunakan oleh empat pengguna dengan otoritas yang berberbeda-beda pada perangkat lunak ini.

Tabel 1. Karakteristik pengguna

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kategori Pengguna** | **Aktivitas** | **Hak** | **Tingkat Pendidikan** | **Keahlian Teknis** |
| Pemilik | * Melakukan pengelolaan karyawan * Melihat semua data | * *Input*, *Edit*, *Delete* dan *View* Karyawan * *View* pemesanan * *View* Pengeluaran | Minimal SMA/SMK/Sederajat | * Dapat mengoperasikan komputer * Dapat menggunakan aplikasi Ms.Office |
| Keuangan | * Melakukan pengelolaan pengeluaran * Melihat dan meng-*edit* data pemesanan * Melihat gaji karyawan | * *Input*, *Edit*, *Delete*, *View* Pengeluaran * *View* Karyawan * *Edit* pemesanan * *View* pemesanan | Minimal SMA/SMK/Sederajat | * Dapat mengoperasikan komputer * Dapat menggunakan aplikasi Ms.Office |
| *Service* | * Melakukan pengelolaan pemesanan * Melihat data produk | * *Input*, *Edit*, *Delete*, *view* pemesanan * *View* Produk | Minimal SMA/SMK/Sederajat | * Dapat mengoperasikan komputer * Dapat menggunakan aplikasi Ms.Office |
| *Staff* Gudang | * Melakukan pengelolaan bahan dan produk * Melihat data pemesanan | * *Input*, *Edit*, *Delete*, *View* Bahan * *Input*, *Edit*, *Delete*, *View* Produk * *View* pemesanan | Minimal SMA/SMK/Sederajat | * Dapat mengoperasikan komputer * Dapat menggunakan aplikasi Ms.Office |

## Lingkungan Operasi

Aplikasi Client server ini akan berfungsi dengan spesifikasi:

* Web Server: Apache
* DBMS: MySQL
* OS: Windows 7 or higher (32-bit or 64-bit)
* Penjelajah situs: *Mozila Firefox, Google Chrome, Opera*
* Bahasa pemograman: HTML, PHP
* Framework : Code Igniter, bootstrap
* Processor: 1 gigahertz (GHz) or faster.
* Memory: 1 GB RAM.
* Graphics card*: Microsoft DirectX 9 graphics device with WDDM driver.*
* Network: Broadband Internet connection.
* Storage: 2 GB available space.

## Batasan Perancangan dan Implementasi

Aplikasi ini hanya untuk pencatatan data produk, bahan, karyawan dan pengeluaran yang ada didalam perusahaan termasuk pembukuan keuangan dimana pengguna hanya dapat meng*input*, menghapus, memperbaharui, dan melihat data-data yang ada di dalam perusahaan. Situs web sistem pembukuan Bakso Najwa ini hanya dapat dibuka pada komputer perusahaan dan hanya admin per divisi yang telah diberikan otoritas yang dapat memiliki akun untuk memakai situs web ini.

## Dokumentasi Pengguna

* *Login Process*
* *How to input* bahan
* *How to input* produk
* *How to input* pengeluaran
* *How to input* pemesanan
* *How to input* karyawan
* *How to edit* bahan
* *How to edit* produk
* *How to edit* pengeluaran
* *How to edit* pemesanan
* *How to edit* karyawan
* *How to edit* pengguna
* *How to delete* bahan
* *How to delete* produk
* *How to delete* pengeluaran
* *How to delete* pemesanan
* *How to delete* karyawan
* *How to view* bahan
* *How to view* produk
* *How to view* pengeluaran
* *How to view* pemesanan
* *How to view* karyawan

## Asumsi dan Dependensi

Asumsi:

1. Hanya admin per divisi yang dapat *login*, sehingga hanya ada empat akun yang dapat *login* termasuk akun pemilik,
2. Produk dibuat ketika pemesan memesan kepada service,
3. Admin *staff* gudang dapat mengelola data bahan dan data produk, *staff* gudang juga dapat melihat data pemesanan,
4. Admin *Service* dapat mengelola data pemesanan dan dapat melihat data produk.
5. Admin Keuangan dapat meng-*edit* dan melihat data pemesanan, mengelola pengeluaran, dan melihat data karyawan.
6. Pemilik dapat melihat semua data yang ada di database*,* dan dapat mengelola data karyawan,
7. Jika seorang *customer* memiliki data status pemesanan belum lunas sebanyak 3 buah maka, customer tidak dibolehkan memesan kecuali status pemesanan sudah lunas,
8. Ketika *Staff* gudang membeli bahan, data stok bahan akan diinputkan ke dalam sistem dan data pembelian bahan (harga bahan) akan diinputkan oleh keuagan sebagai data pengeluaran,

Depedensi:

1. Produk dihasilkan dari Bahan,
2. *Customer* dapat memesan produk,

# Requirements Antarmuka Eksternal

## Antarmuka Pengguna

Pengguna akan berinteraksi dengan situs web sistem pembukuan Bakso Najwa melalui antarmuka dengan penjelajah situs. Pengguna akan membuka penjelajah situs untuk masuk ke situs web sistem pembukuan Bakso Najwa. Untuk menuju menu yang akan di akses diperlukan authentifikasi untuk memproses pengguna tersebut merupakan pemilik, *staff* gudang, *service* atau keuangan. Jika pengguna merupakan pemilik, program web akan menampilkan *menu* berupa *input* karyawan, *edit* karyawan, *delete* karyawan, *view* karyawan, dan *view* pengeluaran. Jika pengguna merupakan *staff* gudang, program web akan menampilkan *menu* berupa *input* bahan, *input* produk, *edit* bahan, *edit* produk, *delete* bahan, *delete* produk, *view* bahan, *view* produk , *view* pemesanan. Jika pengguna merupakan *service*, program web akan menampilkan *menu* berupa *input* pemesanan, *edit* pemesanan, *delete* pemesanan dan *view* pemesanan dan *view* produk. Jika pengguna merupakan keuangan, program web akan menampilkan *menu* berupa *input* pengeluaran, *edit* pengeluaran, *delete* pengeluaran, *edit* pemesanan, *view* pemesanan, dan *view* karyawan. Setelah pengguna terverifikasi, pengguna melakukan fungsionalitas kegiatan yang pada tampilan masing-masing.

## Antarmuka Perangkat Keras

Program web memerlukan perangkat keras berupa keyboard, monitor, tetikus, CPU, dan server untuk menjalankan produk. Untuk perangkat keras seperti keboard, monitor, tetikus dan CPU berfungsi untuk menunjang segala aktifitas yang berkaitan dengan produk. Untuk perangkat keras server berfungsi sebagai sistem operasi jaringan. Server juga menjalankan perangkat lunak administratif yang berfungsi untuk mengontrol akses tiap-tiap komputer terhadap jaringan dan sumber daya yang terdapat didalamnya termasuk database yang dapat memberikan akses kepada workstationya.

## Antarmuka Perangkat Lunak

Produk berupa situs web ini bisa diakses menggunakan segala penjelajah situs (Mozila Firefox, Google Chrome, Opera), dimana web ini dibangun dari bahasa pemrogramanHTML, CSS, PHP dan Javascript. Web ini dapat diakses oleh pengguna web pada lingkungan OS minimal Windows 7 untuk menjalankan segala penjelajah situs yang tersedia di dalam komputer perusahaan.

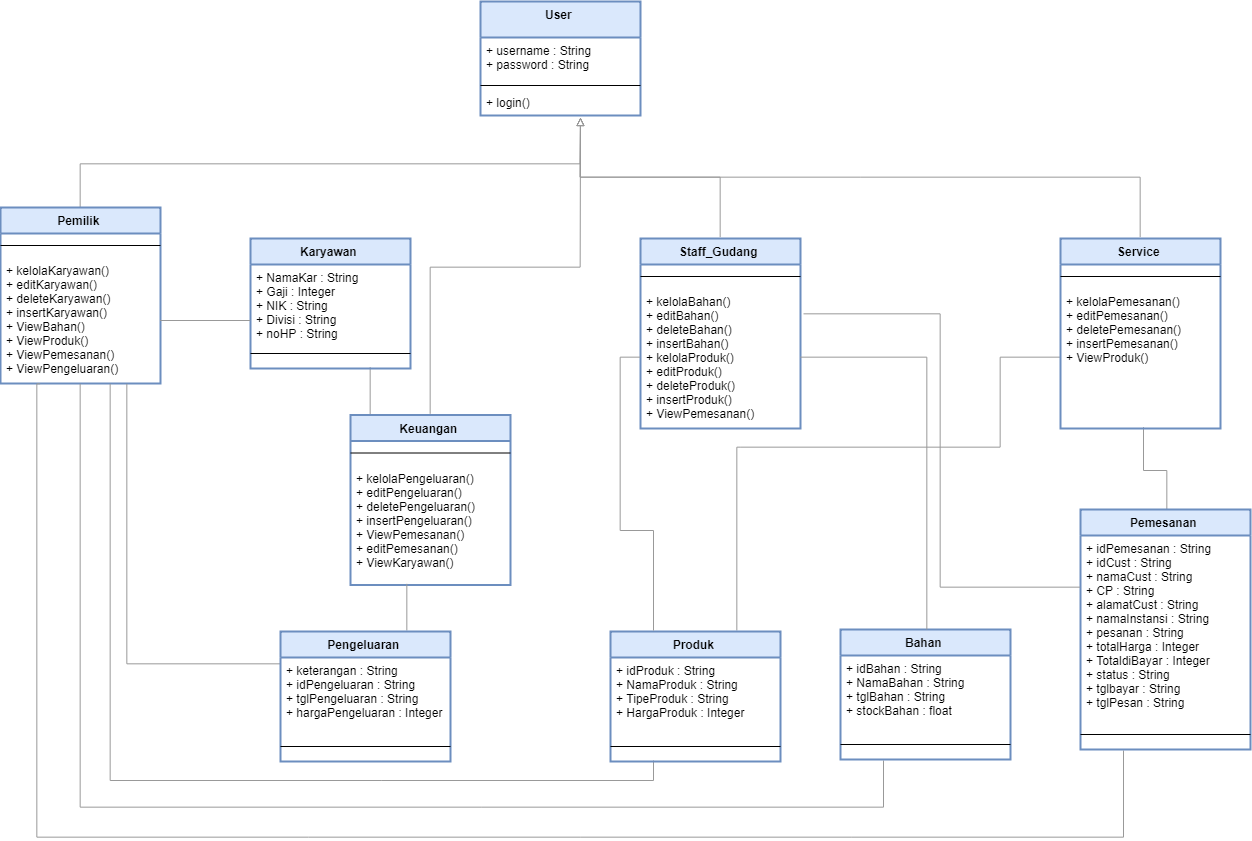
## Antarmuka Komunikasi

Requirements yang dibutuhkan terkait dengan proses komunikasi pada produk ini adalah protokol komunikasi server jaringan lokal. Hal tersebut karena produk bersifat luar jaringan (offline) dan menggunakan database local dan server lokal.

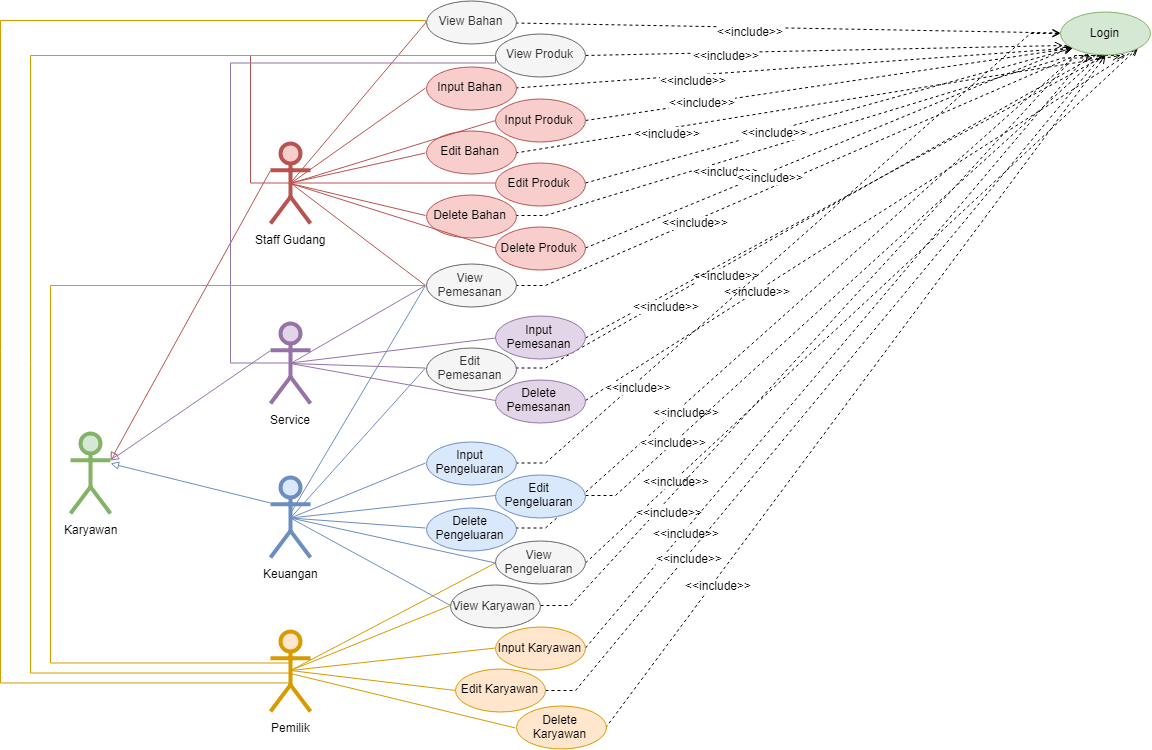
# Model Domain

## Entity Relationship Diagram

## Class Diagram



# Fitur Sistem (Use Cases)



## Use Case *Login*

### Nama Use Case: Login

### Tujuan: Agar aktor (Staff Gudang, Service, Keuangan, Pemilik) dapat menggunakan program web dengan otoritasnya masing masing

### Input: username dan password

### Output: data bahan, data produk, data pemesanan, data pengeluaran, dan data karyawan

### Skenario Utama: Aktor memiliki hak login dan harus melakukan login agar dapat menggunakan program web

### Prakondisi: Aktor ingin menggunakan program web dan belum melakukan login.

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang, *Service*, Keuangan, Pemilik | Sistem |
| * + - 1. *Staff* Gudang/ *Service*/ Keuangan/ Pemilik meng*input*kan username dan password       2. Melakukan *login* |  |
|  | * + - 1. Validasi username dan password dengan data yang sudah tersimpan di database data karyawan atau pemilik       2. Jika tidak sesuai, maka akan tampil notif *login* gagal kembali ke 1       3. Jika sesuai, maka akan tampil notif *login* berhasil lanjut ke 6       4. Menampilkan halaman utama program web dengan otoritasnya masing masing |
| * + - 1. Melakukan aktivitas di dalam program web tersebut. |  |

### Pascakondisi: Aktor telah melakukan login dan siap berinteraksi dengan program web

### Contoh: Pemilik ingin menggunakan program web yang ada, sebelum melakukan itu pemilik harus memasukkan username dan password untuk login agar dapat masuk ke dalam halaman utama program web.

## Use Case *Input* Bahan

### Nama Use Case: Input Bahan

### Tujuan: Untuk mencatat, menambahkan, dan menyimpan data bahan yang baru

### Input: Data bahan (idbahan, nama bahan, jumlah stok bahan, tanggal)

### Output: Data bahan (idbahan, nama bahan, jumlah stok bahan, tanggal) yang tersimpan dalam database bahan

### Skenario Utama: Staff Gudang mempunyai hak input bahan untuk menginput data bahan

### Prakondisi: Jika ingin menggunakan input bahan, Staff Gudang telah login untuk menggunakan menu input bahan dan data bahan belum tersimpan di database bahan

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang | Sistem |
| 1. Membuka *menu* kelola bahan |  |
|  | 1. Menampilkan tampilan kelola bahan |
| 1. Memilih tombol *input* |  |
|  | 1. Menampilkan tampilan *form* *input* |
| 1. Memasukan data bahan 2. Menekan tombol simpan |  |
|  | 1. Jika id tersedia, maka akan mengeluarkan tampilan notifikasi “input failed“ dan kembali ke langkah 2. 2. Jika belum tersedia, maka lanjut ke langkah 9. 3. Merekam data bahan yang di*input*kan *Staff* Gudang 4. Memproses dan menyimpan data bahan ke dalam database bahan 5. Menampilkan data bahan yang telah disimpan |
| 1. Mengecek kembali data bahan yang telah di*input*kan |  |

### Pascakondisi: Staff Gudang telah menginputkan data bahan dan sistem berhasil menyimpan data bahan di database bahan

### Contoh: Staff Gudang ingin input data bahan tepung. Staff Gudang yang merupakan Staff Gudang melakukan input data bahan melalui menu input bahan. Sistem mencatat dan menyimpan kedalam database bahan

## Use Case *View* Bahan

### Nama Use Case: View Bahan

### Tujuan: Untuk melihat data bahan

### Input: idbahan/nama bahan

### Output: Data bahan (idbahan, nama bahan, jumlah stok bahan,tanggal)

### Skenario Utama: Staff Gudang mempunyai hak view bahan untuk melihat data bahan yang tersedia

### Prakondisi: Jika ingin menggunakan view bahan, Staff Gudang telah login untuk menggunakan menu view bahan dan data bahan tersedia di database bahan

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang | Sistem |
| 1. Memilih *menu* kelola bahan |  |
|  | 1. Menampilkan tampilan kelola bahan termasuk tabel data bahan |
| 1. Jika ingin mencari data dari database bahan masukkan idbahan/nama bahan yang ingin dicari dan lanjut ke langkah 5. 2. Jika tidak ingin mencari maka proses melihat data selesai. 3. Tekan tombol search |  |
|  | 1. Mencari data yang di*input*kan dari database bahan 2. Menampilkan data dari database bahan. Jika data tidak di temukan, maka akan menampilkan keterangan data tidak tersedia |

### Pascakondisi: Staff Gudang dapat melihat data bahan di database bahan

### Contoh: Pengguna ingin melihat data bahan tepung. Pengguna yang merupakan Staff Gudang mengecek bahan melalui menu view bahan dan sistem menampilkan data bahan

## Use Case *Edit* Bahan

### Nama Use Case: Edit Bahan

### Tujuan: Memperbaharui data bahan

### Input: Data bahan (idbahan/nama bahan, jumlah stok bahan, tanggal) yang terbaru

### Output: Data bahan terbaru yang tersimpan di dalam database bahan

### Skenario Utama: Staff Gudang mempunyai hak edit bahan untuk memperbaharui data bahan yang tersedia

### Prakondisi: Jika ingin menggunakan edit bahan, Staff Gudang telah login untuk menggunakan menu edit bahan dan data bahan tersedia di database bahan

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang | Sistem |
| 1. Membuka *menu* kelola bahan |  |
|  | 1. Menampilkan tampilan *view* bahan |
| 1. Memasukkan data (nama bahan atau idBahan) yang ingin di-*edit* |  |
|  | 1. Mencari data (namaBahan atau idBahan) yang di*input* oleh *Staff* Gudang 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*edit* 2. Menekan tombol *edit* |  |
|  | 1. Menampilkan *form* *edit* |
| 1. Meng-*edit* data dan menekan tombol simpan |  |
|  | 1. Merekam dan menyimpan data ke dalam database bahan 2. Menampilkan data bahan yang telah diperbaharui |
| 1. Mengecek kembali data bahan yang telah diubah |  |

### Pascakondisi: Staff Gudang telah meng-edit data bahan dan sistem berhasil memperbaharui data bahan di database bahan

### Contoh: Pengguna ingin memperbaharui data bahan tepung. Pengguna yang merupakan Staff Gudang mengecek bahan melalui menu edit bahan dan sistem menampilkan data bahan. Setelah menemukan dan merubah data bahan, sistem memperbaharui data.

## Use Case *Delete* Bahan

### Nama Use Case: Delete Bahan

### Tujuan: Menghapus data bahan

### Input: Id bahan/nama bahan yang akan di delete

### Output: data bahan pada database bahan terbaru

### Skenario Utama: Staff Gudang mempunyai hak delete bahan untuk menghapus data bahan pada database

### Prakondisi: Jika ingin menggunakan delete bahan, Staff Gudang telah login untuk menggunakan menu delete data bahan tersedia di database bahan

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang | Sistem |
| 1. Membuka *menu* kelola bahan |  |
|  | 1. Menampilkan tampilan *view* bahan |
| 1. Memasukkan data (nama bahan atau idBahan) yang ingin dihapus |  |
|  | 1. Mencari data (namaBahan atau idBahan) yang di*input* oleh *Staff* Gudang 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*delete* 2. Menekan tombol *delete* |  |
|  | 1. Menghapus data 2. Memproses dan memperbaharui data database bahan 3. Menampilkan notifikasi “delete success” dan data bahan terbaru |

### Pascakondisi: Staff Gudang telah menghapus data bahan dan sistem berhasil memperbaharui data bahan di database bahan

### Contoh: Staff Gudang menghapus data bahan tepung dari database

## Use Case *Input* Produk

### Nama Use Case: Input Produk

### Tujuan: Untuk mencatat, menambahkan, dan menyimpan data produk yang baru

### Input: Data produk (id produk, nama produk, tanggal, tipe produk, harga produk)

### Output: Data produk yang tersimpan dalam database produk

### Skenario Utama: Staff Gudang mempunyai hak input bahan untuk menginput data produk

### Prakondisi: Jika ingin menggunakan input produk, Staff Gudang telah login untuk menggunakan menu input produk dan data produk belum tersimpan di database produk

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang | Sistem |
| 1. Membuka *menu* kelola produk |  |
|  | 1. Menampilkan tampilan kelola produk |
| 1. Menekan tombol *input* |  |
|  | 1. Menampilkan tampilan *form* *input* |
| 1. Memasukan data produk 2. Menekan tombol simpan |  |
|  | 1. Jika id tersedia, maka akan mengeluarkan tampilan notifikasi “input failed“ dan kembali ke langkah 2. 2. Jika belum tersedia, maka lanjut ke langkah 9. 3. Merekam data produk yang di*input*kan *Staff* Gudang 4. Memproses dan menyimpan data produk ke dalam database produk 5. Menampilkan data produk yang telah disimpan |
| 1. Mengecek kembali data produk yang telah di*input*kan |  |

### Pascakondisi: Staff Gudang telah menginputkan data produk dan sistem berhasil menyimpan data produk di database produk

### Contoh: Pengguna ingin input data produk bakso. Pengguna yang merupakan Staff Gudang melakukan input data produk melalui menu input produk. Sistem mencatat dan menyimpan kedalam database produk

## Use Case *View* Produk

### Nama Use Case: View Produk

### Tujuan: Untuk melihat data bahan

### Input: id produk/nama produk

### Output: Data produk (id produk, nama produk, tanggal, tipe produk, harga produk) yang tersimpan dalam database produk

### Skenario Utama: Staff Gudang atau service mempunyai hak view produk untuk melihat data produk yang tersedia

### Prakondisi: Jika ingin menggunakan view produk, Staff Gudang atau service telah login untuk menggunakan menu view produk dan data produk tersedia di database produk

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang, *service* | Sistem |
| 1. Membuka *menu* kelola produk |  |
|  | 1. Menampilkan tampilan kelola produk termasuk tabel data produk |
| 1. Jika ingin mencari data dari database produk masukkan idproduk/nama produk yang ingin dicari dan lanjut ke langkah 5. 2. Jika tidak ingin mencari maka proses melihat data selesai. 3. Tekan tombol search |  |
|  | 1. Mencari data yang di*input*kan dari database produk 2. Menampilkan data dari database produk. Jika data tidak di temukan, maka akan menampilkan keterangan data tidak tersedia |

### Pascakondisi: Staff Gudang atau service dapat melihat data produk di database produk

### Contoh: Pengguna ingin melihat data produk bakso. Pengguna yang merupakan Staff Gudang atau service mengecek produk melalui menu view produk dan sistem menampilkan data produk

## Use Case *Edit* Produk

### Nama Use Case: Edit Produk

### Tujuan: Memperbaharui data produk

### Input: Data produk (id produk, jumlah stok produk, nama produk, tanggal, tipe produk, harga produk)

### Output: Data produk terbaru yang tersimpan di dalam database produk

### Skenario Utama: Staff Gudang atau service mempunyai hak edit produk untuk memperbaharui data produk yang tersedia

### Prakondisi: Jika ingin menggunakan edit produk, Staff Gudang atau service telah login untuk menggunakan menu edit produk dan data produk tersedia di database produk

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang, *service* | Sistem |
| 1. Membuka *menu* kelola produk |  |
|  | 1. Menampilkan tampilan kelola produk |
| 1. Memasukkan data (nama produk atau idProduk) yang ingin di-*edit* |  |
|  | 1. Mencari data (namaProduk atau idProduk) yang di*input* oleh *Staff* Gudang/*service* 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*edit* 2. Menekan tombol *edit* |  |
|  | 1. Menampilkan *form* *edit* |
| 1. Meng-*edit* data dan menekan tombol simpan |  |
|  | 1. Merekam dan menyimpan data ke dalam database produk 2. Menampilkan data produk yang telah diperbaharui |
| 1. Mengecek kembali data produk yang telah diubah |  |

### Pascakondisi: Staff Gudang atau service telah meng-edit data produk dan sistem berhasil memperbaharui data produk di database produk

### Contoh: Pengguna ingin memperbaharui data produk bakso. Pengguna yang merupakan Staff Gudang atau service mengecek produk melalui menu edit produk dan sistem menampilkan data produk Setelah menemukan dan merubah data produk, sistem memperbaharui data.

## Use Case *Delete* Produk

### Nama Use Case: Delete Produk

### Tujuan: Menghapus data produk

### Input: data produk (idproduk/nama produk) yang akan di delete

### Output: data produk pada database produk terbaru

### Skenario Utama: Staff Gudang mempunyai hak delete bahan untuk menghapus data produk pada database

### Prakondisi: Jika ingin menggunakan delete produk, Staff Gudang telah login untuk menggunakan menu delete data produk tersedia di database produk

### Langkah-langkah:

|  |  |
| --- | --- |
| *Staff* Gudang | Sistem |
| 1. Membuka *menu* kelola produk |  |
|  | 1. Menampilkan tampilan kelola produk |
| 1. Memasukkan data (nama produk atau idProduk) yang ingin dihapus |  |
|  | 1. Mencari data (namaProduk atau idProduk) yang di*input* oleh *Staff* Gudang 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*delete* 2. Menekan tombol *delete* |  |
|  | 1. Menghapus data 2. Memproses dan memperbaharui data database produk 3. Menampilkan notifikasi “delete success” dan data produk terbaru |

### Pascakondisi: Staff Gudang telah menghapus data produk dan sistem berhasil memperbaharui data bahan di database produk

### Contoh: Pengguna menghapus data produk bakso dari database

## Use Case *Input* Pemesanan

### Nama Use Case: input pemesanan

### Tujuan: Untuk mencatat, menambahkan, dan menyimpan data pemesanan yang baru

### Input: data pemesanan (nama konsumen, nama instansi, kontak personal, alamat, pesanan, total harga pemesanan, status pembayaran, tanggal pemesanan, tanggal pembayaran, id pemesanan)

### Output: data pemesanan yang tersimpan dalam database pemesanan

### Skenario Utama: Service mempunyai hak input pemesanan untuk menginput data pemesanan

### Prakondisi: Jika ingin menggunakan input pemesanan, Service telah login untuk menggunakan menu input pemesanan dan data pemesanan belum tersimpan di database pemesanan

### Langkah-langkah:

|  |  |
| --- | --- |
| *Service* | Sistem |
| 1. Membuka *menu* kelola pemesanan |  |
|  | 1. Menampilkan tampilan kelola pemesanan |
| 1. Menekan tombol *input* |  |
|  | 1. Menampilkan tampilan *form* *input* |
| 1. Memasukan data pemesanan 2. Menekan tombol simpan |  |
|  | 1. Jika id tersedia, maka akan mengeluarkan tampilan notifikasi “input failed“ dan kembali ke langkah 2. 2. Jika belum tersedia, maka lanjut ke langkah 9. 3. Merekam data pemesanan yang di*input*kan *Service* 4. Memproses dan menyimpan data pemesanan ke dalam database pemesanan 5. Menampilkan data pemesanan yang telah disimpan |
| 1. Mengecek kembali data pemesanan yang telah di*input*kan |  |

### Pascakondisi: Service telah menginputkan data pemesanan dan sistem berhasil menyimpan data pemesanan di database pemesanan

### Contoh: Pengguna ingin input data pemesanan bakso. Pengguna yang merupakan Service melakukan input data pemesanan melalui menu input pemesanan. Sistem mencatat dan menyimpan kedalam database pemesanan

## Use Case *View* Pemesanan

### Nama Use Case: view pemesanan

### Tujuan: Untuk melihat data pemesanan

### Input: idPemesanan/nama konsumen/tanggal pemesanan

### Output: data pemesanan

### Skenario Utama: Service, Staff Gudang, Keuangan atau Pemilik mempunyai hak view pemesanan untuk melihat data pemesanan yang tersedia

### Prakondisi: Jika ingin menggunakan view pemesanan, Service, Staff Gudang, Keuangan atau Pemilik telah login untuk menggunakan menu view pemesanan dan data pemesanan tersedia di database pemesanan

### Langkah-langkah:

|  |  |
| --- | --- |
| *Service*, *Staff* Gudang, Keuangan atau Pemilik | Sistem |
| 1. Membuka *menu* kelola pemesanan |  |
|  | 1. Menampilkan tampilan kelola pemesanan termasuk tabel data pemesanan |
| 1. Jika ingin mencari data dari database pemesanan masukkan idPemesanan/nama konsumen/tanggal pemesanan yang ingin dicari dan lanjut ke langkah 5. 2. Jika tidak ingin mencari maka proses melihat data selesai. 3. Tekan tombol search |  |
|  | 1. Mencari data yang di*input*kan dari database pemesanan 2. Menampilkan data dari database pemesanan. Jika data tidak di temukan, maka akan menampilkan “Data tidak tersedia” |

### Pascakondisi: Service, Staff Gudang, Keuangan atau Pemilik telah melihat data pemesanan

### Contoh: Pengguna ingin melihat data pemesanan. Pengguna yang merupakan Service, Staff Gudang, Keuangan atau Pemilik mengecek data pemesanan melalui menu view pemesanan dan sistem menampilkan data pemesanan

## Use Case *Edit* Pemesanan

### Nama Use Case: edit pemesanan

### Tujuan: Memperbaharui data pemesanan

### Input: data pemesanan (nama konsumen, nama instansi, kontak personal, alamat, pesanan, total harga pemesanan, status pembayaran, tanggal pemesanan, tanggal pembayaran,id pemesanan)

### Output: data pemesanan terbaru yang tersimpan di dalam database pemesanan

### Skenario Utama: Service atau Keuangan mempunyai hak edit pemesanan untuk memperbaharui data pemesanan yang tersedia

### Prakondisi: Jika ingin menggunakan edit pemesanan, Service atau Keuangan telah login untuk menggunakan menu edit pemesanan dan data pemesanan tersedia di database pemesanan

### Langkah-langkah:

|  |  |
| --- | --- |
| *Service*,Keuangan | Sistem |
| 1. Membuka *menu* kelola pemesanan |  |
|  | 1. Menampilkan tampilan kelola pemesanan |
| 1. Memasukkan data (nama konsumen atau idPemesanan) yang ingin di-*edit* |  |
|  | 1. Mencari data (nama konsumen atau idPemesanan) yang di*input* oleh Keuangan/*service* 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*edit* 2. Menekan tombol *edit* |  |
|  | 1. Menampilkan *form* *edit* |
| 1. Meng-*edit* data dan menekan tombol simpan |  |
|  | 1. Merekam dan menyimpan data ke dalam database pemesanan 2. Menampilkan data pemesanan yang telah diperbaharui |
| 1. Mengecek kembali data pemesanan yang telah diubah |  |

### Pascakondisi: Service atau Keuangan telah meng-edit data pemesanan dan sistem berhasil memperbaharui data pemesanan di database pemesanan

### Contoh: Pengguna ingin memperbaharui data pemesanan bakso. Pengguna yang merupakan Service dan Keuangan mengecek pemesanan melalui menu edit pemesanan dan sistem menampilkan data pemesanan. Setelah menemukan dan merubah data pemesanan, sistem memperbaharui data.

## Use Case *Delete* Pemesanan

### Nama Use Case: delete pemesanan

### Tujuan: Menghapus data pemesanan

### Input: data pemesanan (id pemesanan/nama konsumen) yang akan di delete

### Output: data pemesanan terbaru

### Skenario Utama: Service mempunyai hak delete pemesanan untuk menghapus data pemesanan pada database

### Prakondisi: Jika ingin menggunakan delete pemesanan, Service telah login untuk menggunakan menu delete pemesanan dan data pemesanan tersedia di database pemesanan.

### Langkah-langkah:

|  |  |
| --- | --- |
| *Service* | Sistem |
| 1. Membuka *menu* kelola pemesanan |  |
|  | 1. Menampilkan tampilan kelola pemesanan |
| 1. Memasukkan data (nama konsumen atau idPemesanan) yang ingin dihapus |  |
|  | 1. Mencari data (nama konsumen atau idPemesanan) yang di*input* oleh *Service* 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*delete* 2. Menekan tombol *delete* |  |
|  | 1. Menghapus data 2. Memproses dan memperbaharui data database pemesanan 3. Menampilkan notifikasi “delete success” dan data pemesanan terbaru |

### Pascakondisi: Service telah menghapus data pemesanan dan sistem berhasil memperbaharui data pemesanan di database pemesanan

### Contoh: Service menghapus data pemesanan bakso dari database pemesanan

## Use Case *Input* Pengeluaran

### Nama Use Case: Input Pengeluaran

### Tujuan: Untuk mencatat, menambahkan, dan menyimpan data pengeluaran yang baru

### Output: Data pengeluaran yang tersimpan dalam database pengeluaran

### Input: data pengeluaran (keterangan, harga pengeluaran, tanggal pengeluaran)

### Skenario Utama: Keuangan mempunyai hak input pengeluaran untuk menginput data pengeluaran

### Prakondisi: Jika ingin menggunakan input pengeluaran, Keuangan telah login untuk menggunakan menu input pengeluaran dan data pengeluaran belum tersimpan di database pengeluaran

### Langkah-langkah:

|  |  |
| --- | --- |
| Keuangan | Sistem |
| 1. Membuka *menu* kelola pengeluaran |  |
|  | 1. Menampilkan tampilan kelola pengeluaran |
| 1. Menekan tombol *input* |  |
|  | 1. Menampilkan tampilan *form input* |
| 1. Memasukan data pengeluaran 2. Menekan tombol simpan |  |
|  | 1. Merekam data pengeluaran yang di*input*kan Keuangan 2. Memproses dan menyimpan data pengeluaran ke dalam database pengeluaran 3. Menampilkan data pengeluaran yang telah disimpan |
| 1. Mengecek kembali data pengeluaran yang telah di*input*kan |  |

### Pascakondisi: Keuangan telah menginputkan data pengeluaran dan sistem berhasil menyimpan data pengeluaran di database pengeluaran

### Contoh: Pengguna ingin input data pengeluaran listrik. Pengguna yang merupakan Keuangan melakukan input data pengeluaran melalui menu input pengeluaran. Sistem mencatat dan menyimpan kedalam database pengeluaran

## Use Case *View* Pengeluaran

### Nama Use Case: View Pengeluaran

### Tujuan: Untuk melihat data pengeluaran

### Input: idPengeluaran/ keterangan/ tanggal pengeluaran

### Output: Untuk melihat data pengeluaran

### Skenario Utama: Keuangan atau Pemilik mempunyai hak view pengeluaran untuk melihat data pengeluaran yang tersedia

### Prakondisi: Jika ingin menggunakan view pengeluaran, Keuangan atau Pemilik telah login untuk menggunakan menu view pengeluaran dan data pengeluaran tersedia di database pengeluaran

### Langkah-langkah:

|  |  |
| --- | --- |
| Keuangan, Pemilik | Sistem |
| 1. Membuka *menu* kelola pengeluaran |  |
|  | 1. Menampilkan tampilan kelola pengeluaran termasuk tabel data pengeluaran |
| 1. Jika ingin mencari data dari database pengeluaran masukkan id pengeluaran/keterangan/tanggal pemesanan yang ingin dicari dan lanjut ke langkah 5. 2. Jika tidak ingin mencari maka proses melihat data selesai. 3. Tekan tombol search |  |
|  | 1. Mencari data yang di*input*kan dari database pengeluaran 2. Menampilkan data dari database pengeluaran. Jika data tidak di temukan, maka akan menampilkan keterangan data tidak tersedia |

### Pascakondisi: Keuangan atau Pemilik telah melihat data pengeluaran

### Contoh: Pengguna ingin melihat data pengeluaran. Pengguna yang merupakan Keuangan atau Pemilik mengecek data pengeluaran apa saja melalui menu view pengeluaran dan sistem menampilkan data pengeluaran

## Use Case *Edit* Pengeluaran

### Nama Use Case: Edit Pengeluaran

### Tujuan: Memperbaharui data pengeluaran

### Input: Data pengeluaran (id pengeluaran, keterangan, harga pengeluaran, tanggal pengeluaran) yang terbaru

### Output: Data pengeluaran terbaru yang tersimpan di dalam database pengeluaran

### Skenario Utama: Keuangan mempunyai hak edit pengeluaran untuk memperbaharui data pengeluaran yang tersedia

### Prakondisi: Jika ingin menggunakan edit pengeluaran, Keuangan telah login untuk menggunakan menu edit pengeluaran dan data pengeluaran tersedia di database pengeluaran

### Langkah-langkah:

|  |  |
| --- | --- |
| Keuangan | Sistem |
| 1. Membuka *menu* kelola pengeluaran |  |
|  | 1. Menampilkan tampilan kelola pengeluaran |
| 1. Memasukkan data (keterangan atau idPengeluaran) yang ingin di-*edit* |  |
|  | 1. Mencari data (keterangan atau idPengeluaran) yang di*input* oleh Keuangan 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*edit* 2. Menekan tombol *edit* |  |
|  | 1. Menampilkan *form* *edit* |
| 1. Meng-*edit* data dan menekan tombol simpan |  |
|  | 1. Merekam dan menyimpan data ke dalam database pengeluaran 2. Menampilkan data pengeluaran yang telah diperbaharui |
| 1. Mengecek kembali data pengeluaran yang telah diubah |  |

### Pascakondisi: Keuangan telah meng-edit data pengeluaran dan sistem berhasil memperbaharui data pengeluaran di database pengeluaran

### Contoh: Pengguna ingin memperbaharui data pengeluaran listrik. Pengguna yang merupakan Keuangan mengecek pengeluaran melalui menu edit pengeluaran dan sistem menampilkan data pengeluaran. Setelah menemukan dan merubah data pengeluaran, sistem memperbaharui data.

## Use Case *Delete* Pengeluaran

### Nama Use Case: Delete Pengeluaran

### Tujuan: Menghapus data pengeluaran

### Input: Data pengeluaran (keterangan) yang akan di delete

### Output: Data pengeluaran terbaru

### Skenario Utama: Keuangan mempunyai hak delete pengeluaran untuk menghapus data pengeluaran pada database

### Prakondisi: Jika ingin menggunakan delete pengeluaran, Keuangan telah login untuk menggunakan menu delete pengeluaran dan data pengeluaran tersedia di database tambahan

### Langkah-langkah:

|  |  |
| --- | --- |
| Keuangan | Sistem |
| 1. Membuka *menu* kelola pengeluaran |  |
|  | 1. Menampilkan tampilan kelola pengeluaran |
| 1. Memasukkan data (nama konsumen atau idPengeluaran) yang ingin dihapus |  |
|  | 1. Mencari data (nama konsumen atau idPengeluaran) yang di*input* oleh Keuangan 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*delete* 2. Menekan tombol *delete* |  |
|  | 1. Menghapus data 2. Memproses dan memperbaharui data database pengeluaran 3. Menampilkan notifikasi “delete success” dan data pengeluaran terbaru |

### Pascakondisi: Keuangan telah menghapus data pengeluaran dan sistem berhasil memperbaharui data pengeluaran di database pengeluaran

### Contoh: Keuangan menghapus data pengeluaran dari database

## Use Case *Input* Karyawan

### Nama Use Case: Input Karyawan

### Tujuan: Untuk menambahkan data karyawan yang baru

### Input: data karyawan (Nama karyawan, gaji, NIK, divisi, nomor HP)

### Output: Data karyawan yang tersimpan dalam database karyawan

### Skenario Utama: Pemilik mempunyai hak input karyawan untuk menginput data karyawan

### Prakondisi: Jika ingin menggunakan input karyawan, Pemilik telah login untuk menggunakan menu input karyawan dan data karyawan belum tersimpan di database karyawan

### Langkah-langkah:

|  |  |
| --- | --- |
| Pemilik | Sistem |
| 1. Membuka *menu* kelola karyawan |  |
|  | 1. Menampilkan tampilan kelola karyawan |
| 1. Menekan tombol *input* |  |
|  | 1. Menampilkan tampilan *form* *input* |
| 1. Memasukan data karyawan 2. Menekan tombol simpan |  |
|  | 1. Jika id tersedia, maka akan mengeluarkan tampilan notifikasi “input failed“ dan kembali ke langkah 2. 2. Jika belum tersedia, maka lanjut ke langkah 9. 3. Merekam data karyawan yang di*input*kan Pemilik 4. Memproses dan menyimpan data karyawan ke dalam database karyawan 5. Menampilkan data karyawan yang telah disimpan |
| 1. Mengecek kembali data karyawan yang telah di*input*kan |  |

### Pascakondisi: Pemilik telah menginputkan data karyawan dan sistem berhasil menyimpan data karyawan di database karyawan

### Contoh: Pengguna ingin input data karyawan baru. Pengguna yang merupakan Pemilik melakukan input data karyawan melalui menu input karyawan. Sistem mencatat dan menyimpan kedalam database karyawan

## Use Case *View* Karyawan

### Nama Use Case: View Karyawan

### Tujuan: Untuk melihat data karyawan

### Input: data karyawan (Nama Karyawan/NIK)

### Output: Data Karyawan

### Skenario Utama: Pemilik atau keuangan mempunyai hak view karyawan untuk melihat data biaya karyawan yang tersedia

### Prakondisi: Jika ingin menggunakan view karyawan, Pemilik atau keuangan telah login untuk menggunakan menu view data karyawan dan data karyawan tersedia di database karyawan

### Langkah-langkah:

|  |  |
| --- | --- |
| Keuangan, Pemilik | Sistem |
| 1. Membuka *menu* kelola karyawan |  |
|  | 1. Menampilkan tampilan kelola karyawan termasuk tabel data karyawan |
| 1. Jika ingin mencari data dari database karyawan masukkan NIK/nama karyawan yang ingin dicari dan lanjut ke langkah 5. 2. Jika tidak ingin mencari maka proses melihat data selesai. 3. Tekan tombol search |  |
|  | 1. Mencari data yang di*input*kan dari database karyawan 2. Menampilkan data dari database karyawan. Jika data tidak di temukan, maka akan menampilkan keterangan data tidak tersedia |

### Pascakondisi: Pemilik atau keuangan telah melihat data karyawan

### Contoh: Pemilik atau keuangan ingin melihat data karyawan. Pemilik atau keuangan mengecek karyawan siapa saja melalui tabel view karyawan dan sistem menampilkan data karyawan

## Use Case *Edit* Karyawan

### Nama Use Case: Edit Karyawan

### Tujuan: Memperbaharui data karyawan

### Input: Data karyawan yang terbaru (Nama karyawan, gaji, NIK, divisi, nomor HP)

### Output: Data karyawan terbaru yang tersimpan di dalam database karyawan

### Skenario Utama: Pemilik mempunyai hak edit karyawan untuk memperbaharui data karyawan yang tersedia

### Prakondisi: Jika ingin menggunakan edit karyawan, Pemilik telah login untuk menggunakan menu edit karyawan dan data karyawan tersedia di database karyawan

### Langkah-langkah:

|  |  |
| --- | --- |
| Pemilik | Sistem |
| 1. Membuka *menu* kelola karyawan |  |
|  | 1. Menampilkan tampilan kelola karyawan |
| 1. Memasukkan data (nama karyawan/NIK) yang ingin di-*edit* |  |
|  | 1. Mencari data (nama karyawan/NIK) yang di*input* oleh Pemilik 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*edit* 2. Menekan tombol *edit* |  |
|  | 1. Menampilkan *form* *edit* |
| 1. Meng-*edit* data dan menekan tombol simpan |  |
|  | 1. Merekam dan menyimpan data ke dalam database karyawan 2. Menampilkan data karyawan yang telah diperbaharui |
| 1. Mengecek kembali data karyawan yang telah diubah |  |

### Pascakondisi: Pemilik telah meng-edit data karyawan dan sistem berhasil memperbaharui data karyawan di database karyawan

### Contoh: Pengguna ingin memperbaharui data karyawan X. Pengguna yang merupakan Pemilik mengecek karyawan melalui menu edit karyawan dan sistem menampilkan data karyawan. Setelah menemukan dan merubah data karyawan, sistem memperbaharui data.

## Use Case *Delete* Karyawan

### Nama Use Case: Delete Karyawan

### Tujuan: Menghapus data karyawan

### Input: Data karyawan yang akan di delete (nama karyawan/NIK)

### Output: Data karyawan terbaru

### Skenario Utama: Pemilik mempunyai hak delete karyawan untuk menghapus data karyawan pada database

### Prakondisi: Jika ingin menggunakan delete karyawan, Pemilik telah login untuk menggunakan menu delete karyawan dan data karyawan tersedia di database karyawan

### Langkah-langkah:

|  |  |
| --- | --- |
| Pemilik | Sistem |
| 1. Membuka *menu* kelola karyawan |  |
|  | 1. Menampilkan tampilan kelola karyawan |
| 1. Memasukkan data (nama karyawan/NIK) yang ingin dihapus |  |
|  | 1. Mencari data (nama karyawan/NIK) yang di*input* oleh Pemilik 2. Jika data yang di cari di temukan, maka akan menampilkan data, lanjutkan ke langkah nomor 7. 3. Jika data yang di cari tidak di temukan, maka akan menampilkan keterangan data tidak tersedia dan kembali ke langkah nomor dua |
| 1. Memilih data yang akan di-*delete* 2. Menekan tombol *delete* |  |
|  | 1. Menghapus data 2. Memproses dan memperbaharui data database karyawan 3. Menampilkan notifikasi “delete success” dan data karyawan terbaru |

### Pascakondisi: Pemilik telah menghapus data karyawan dan sistem berhasil memperbaharui data karyawan di database karyawan.

### Contoh: Pemilik menghapus data karyawan dari database

# Requirements Nonfungsional Lainnya

## Requirements Performa

Web sistem pembukuan Bakso Najwa mampu mengolah data pengeluaran, bahan, produk, pemesanan, dan data karyawan setiap harinya. Situs web ini dapat diakses pada PC/komputer berbeda di perusahaan tetapi dengan server yang sama.

## Requirements Keselamatan

Untuk melindungi kehilangan data, segala data yang ada pada situs web ini akan disimpan ke database dan server lokal dan situs web ini tidak mengizinkan sembarang karyawan memiliki akses ke database tertentu. Divisi *staff*-gudang hanya dapat mengakses database dari bahan dan produk, begitu pula dengan *service*, keuangan dan pemilik sendiri. Selain itu juga data akan dibackup minimal setiap hari agar menghindari resiko kehilangan data.

## Requirements Keamanan

Untuk melindungi perangkat lunak dan keamanan data di dalam DBMS dari akses ilegal, pengubahan dan pengerusakan dari pihak yang tidak berwenang maka dibuat suatu *login* pengguna yang mana akan lebih memilah data akses agar database yang ada tidak bisa diinterferensi dengan akses lain.

## Atribut Kualitas Perangkat Lunak

### Ketersediaan (Availability)

Perangkat lunak ini dapat diakses 24 jam 7 hari.

### Keandalan

Faktor-faktor yang dibutuhkan untuk mencapai keandalan sistem adalah

Perangkat lunak maupun perangkat keras telah lulus uji dalam ketahanan.

Dibutukan sumber daya manusia yang memiliki pengalaman dalam penggunaan sistem yang sejenis agar dapat mencapai keandalan *system.*

### Kepeliharaan (maintainability)

Untuk *maintainability* perangkat lunak kompleksitasnya mudah karena bahasa pemrograman yang digunakan adalah HTML, CSS, Javascript. Kami juga menggunakan framework bootstrap dan codeIgniter agar memudahkan developer dalam proses pengembangan.

# Requirements Lain

Requirement lain yang diperlukan dalam perancangan sistem pembukuan bakso najwa adalah sebagai berikut:

DBMS yang akan dipakai adalah mySQL

Web Server yang digunakan adalah appache yang terintegrasi dengan mySQL

Lampiran A: Daftar Kata-Kata Sukar

Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini terdapat beberapa istilah diantaranya:

* Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan dokumen hasil analisis yang berisi spesifikasi kebutuhan pengguna
* Login merupakan proses autentifikasi pengguna sebelum menggunakan suatu jaringan system
* Logout merupakan proses keluar atau memutus akses dari jaringan system
* Username adalah alamat email yang diregistrasikan pada suatu jaringan sistem tertentu
* Password adalah kumpulan karakter yang digunakan oleh pengguna jaringan untuk memverifikasi identitas dirinya kepada sistem keamanan yang dimiliki oleh jaringan atau sistem tersebut.
* Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam
* Use Case Diagram adalah suatu diagram yang mendeskripsikan fungsionalitas suatu system berdasarkan sudut pandang pengguna
* Use Case Skenario adalah alur jalannya proses use case dari sisi use case dan system
* Flowmap adalah gabungan peta dan flowchart yang menunjukan aliran dokumen dari suatu tempat ke tempat lain
* Flowchart adalah sutu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail
* Entity Relationship Diagram (ERD) merupakan suatu model untuk menjelaskan hubungan antardata dalam database berdasarkan objek-objek dasar data yang mempunyai hubungan antarrelasi
* Database adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program web komputer untuk memperoleh informasi dari basis data tersebut
* Requirement adalah gambaran dari layanan (services) dan batasan bagi sistem yang akan dibangun
* Database Management System (DBMS) adalah suatu sistem atau perangkat lunak yang dirancang untuk mengelola suatu basis data dan menjalankan operasi terhadap data yang diminta banyak pengguna

Lampiran B: Analysis Models

* Proses Bisnis

Bapak Nikman adalah seorang pemilik dari perusahaan Bakso Najwa yang bergerak pada bidang makanan, khususnya bakso ikan. Perusahaan ini terletak di jalan Caringin, Bandung. Beliau merintis usahanya sejak tahun 2016. Terdapat empat divisi, yaitu *Service*, *Staff* Gudang, *Staff* Produksi, dan Keuangan, dimana setiap divisi memiliki satu orang yang bertanggungjawab.

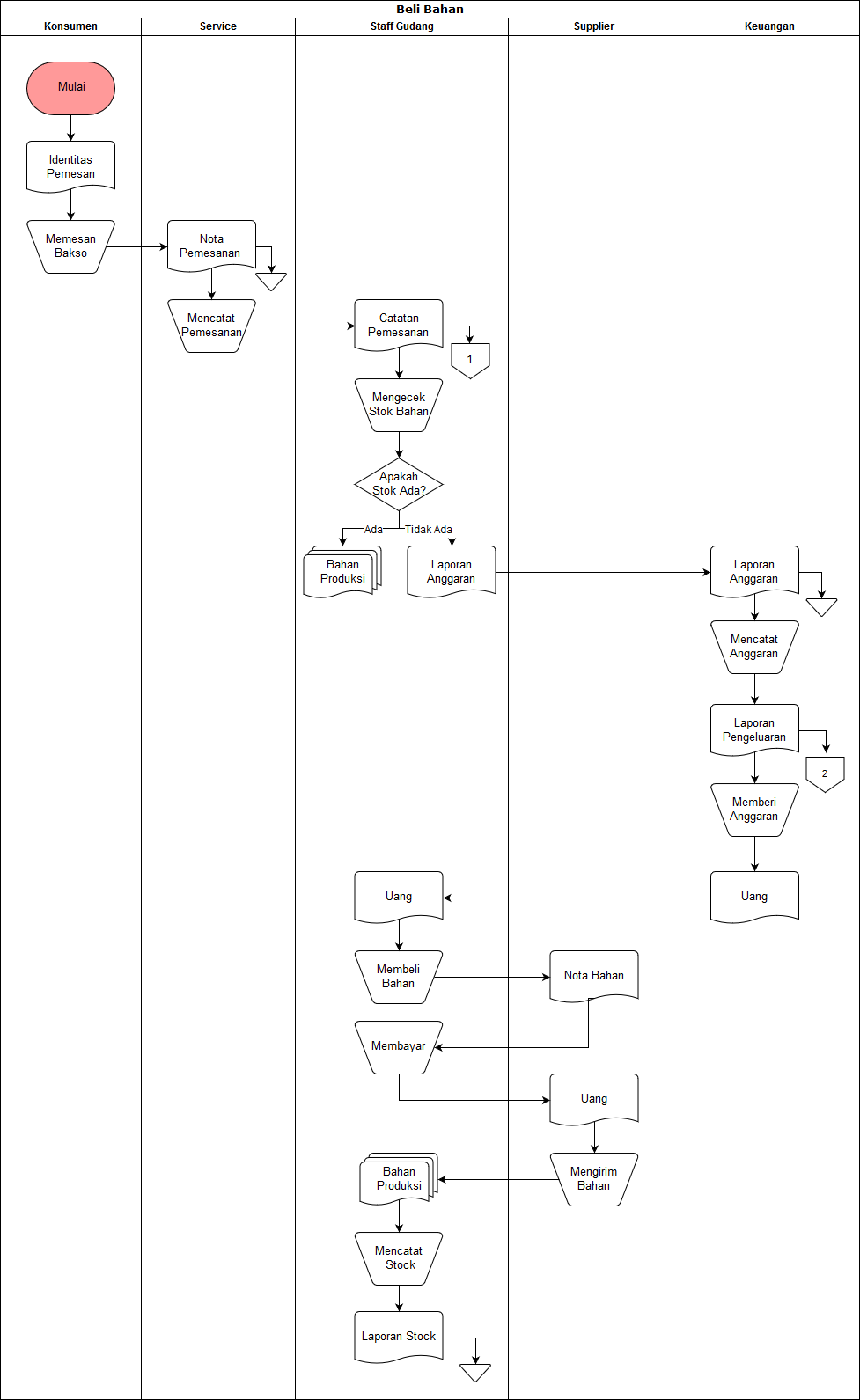
*Service* bertugas untuk melayani pemesan/konsumen, mencatat pesanan, mengantarkan produk, menerima dan memberikan uang dari pemesan ke bagian Keuangan. *Staff* Gudang bertugas mengelola (mencatat dan mengecek) bahan dan produk seperti jumlah stok dan laporan. *Staff* Produksi bertugas membuat dan mengemas bakso. Keuangan bertugas mengelola pengeluaran dan pemasukan, memiliki hak memberi dana kepada divisi lain, memvalidasi jumlah uang yang di terima dari *Service*, membuat laporan keuangan ke Pemilik.

Di dalam proses bisnis Bakso Najwa terdapat tiga proses utama, yaitu proses pembelian bahan baku, proses produksi, dan proses pengiriman produk ke konsumen. Dalam proses pembelian bahan baku, sebelumnya *Service* menerima telepon dari konsumen lalu mecatat pesanan tersebut (nama konsumen, nama instansi, alamat, kontak personal, jumlah pesanan, total harga pengeluaran, tanggal pengeluaran). Catatan pengeluaran tersebut diberikan kepada *Staff* Gudang dan ia akan mengecek apakah bahan produksi tersedia atau tidak. Jika tidak tersedia, *Staff* Gudang akan membuat laporan anggaran pengeluaran bahan ke Keuangan. Setelah Keuangan menyetujui dan memberi dana, *Staff* Gudang membeli bahan dari beberapa tempat, diantaranya membeli ikan segar di Alfa Dinar (perusahaan yang bergerak di bidang bahan baku bakso ikan), membeli bumbu rempah-rempah, tepung dan es di Pasar Caringin. Bahan baku baru akan dicatat seperti nama bahan, jumlah stok bahan, tanggal, dan harga bahan.

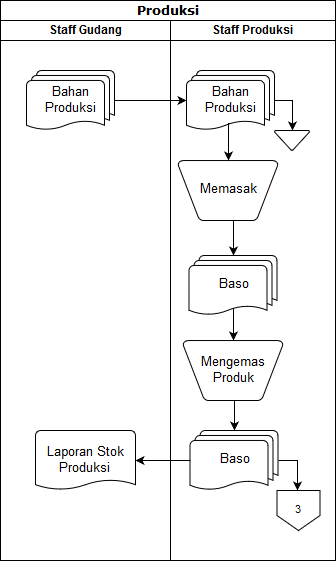
Setelah bahan baku tersedia, maka dilakukan proses pembuatan produk oleh *Staff* Produksi meliputi menggiling ikan dan es dengan mesin penggiling, lalu meracik bumbu dan rempah-rempah, kemudian mencampurkan ikan yang sudah di giling, tepung, dan bumbu, mencetak dan merebus bakso, meniriskan dan mengemas bakso. Setelah selesai di kemas, produk ini akan dicatat terlebih dahulu nama produk, jumlah stok produk, tanggal, tipe produk, dan harga produk, lalu dicek dan diberi nota pengeluaran dan dikirimkan ke konsumen yang memesan oleh *Service*.

Ketika produk sudah sampai ke tangan konsumen, konsumen dapat membayarnya secara cash/kredit melalui *Service*. *Service* akan memberikan rekapan nota pengeluaran dan uang dari konsmen ke Keuangan. Jika konsumen membayar secara kr*edit*, *Service* akan memberikan nota kr*edit* ke Keuangan. Biaya pengeluaran (keterangan pengeluaran, harga pengeluaran, tanggal pengeluaran) dicatat dalam pembukuan perusahaan oleh Keuangan dan dilaporkan pada Pemilik. Namun, dalam pencatatan ini perusahaan memiliki kendala, seringkali terdapat nota pembelian yang hilang atau tercecer, serta belum ada pembukuan pemasukan dan pengeluaran yang tersimpan dengan baik.

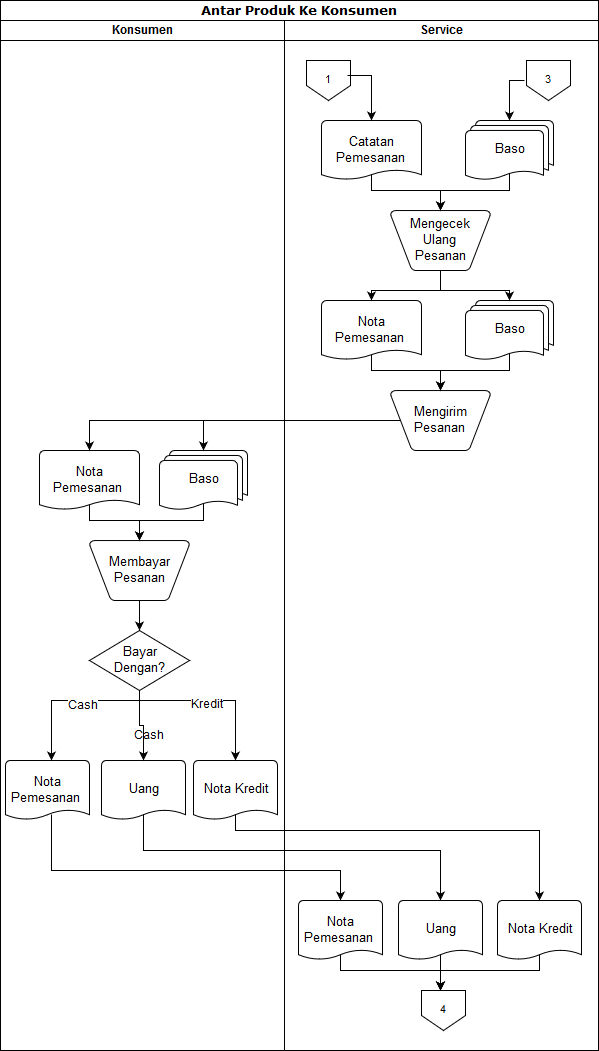
* Aturan Bisnis
* Pemilik dapat merekrut karyawan secara langsung, mengelola karyawan dan melihat laporan setiap divisi
* *Service* dapat mengantarkan produk ke banyak customer
* *Service* dapat mengelola banyak pemesan dan pesanannya
* *Staff* Gudang dapat mengelola banyak bahan
* *Staff* Gudang dapat mengelola banyak produk
* *Staff* Produksi dapat membuat dan mengemas banyak produk
* Keuangan dapat mengelola banyak pengeluaran
* Pemilik dapat melihat mengelola/merekrut banyak karyawan
* Konsumen dapat membayar secara cash/kr*edit*
* Konsumen hanya dibolehkan untuk mendapatkan tiga nota dengan status pembayaran kr*edit*
* Flowmap
* Flowmap Beli Bahan



* Flowmap Produksi



* Flowmap Antar Produk ke Konsumen



* Flowmap Laporan Hasil Penjualan

