*Software Requirements Specification*

for

Aplikasi Kasir dan Manajemen Stok – Warkop Bunny

Version 1.0 approved

Prepared by

1122140087 – Bintang Kharisma

1122140045 - Bintang Poetra Al Fattah

1122140110 - Dika Yonanda Putra

23 Mei 2025

Table of Contents

1. Pendahuluan 1

1.1 Tujuan Penulisan Dokumen 1

1.2 Audien yang Dituju dan Pembaca yang Disarankan 1

1.3 Batasan Produk 1

1.4 Definisi dan Istilah 1

1.5 Refrensi 1

2. Deskripsi Keseluruhan 2

2.1 Deskripsi Produk 2

2.2 Fungsi Produk 2

2.3 Penggolongan Karakterik Pengguna 2

2.4 Lingkungan Operasi 2

2.5 Batasan Desain dan Implementasi 2

2.6 Dokumentasi Pengguna 3

3. Kebutuhan Antarmuka Eksternal 4

3.1 User Interfaces 4

3.2 Hardware Interface 4

3.3 Software Interface 4

3.4 Communication Interface 4

4. Functional Requirement 5

4.1 Use Case Diagram 5

4.2 Nama Use Case 1 5

4.3 Nama Use Case 2 5

4.4 Class Diagram 6

5. Non Functional Requirements 7

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini disusun untuk menjabarkan kebutuhan fungsional dan non-fungsional dari sistem aplikasi **pencatatan penjualan** dan **manajemen stok barang** pada **Warkop Bunny.** Aplikasi ini akan dikembangkan menggunakan Flutter sebagai kerangka kerja frontend dan **Firebase/mysql** sebagai layanan backend. Dengan dokumen ini, pengembang dapat memiliki acuan sistematis dalam membangun aplikasi, sedangkan pemilik usaha dan staf operasional dapat memahami ruang lingkup serta fitur yang tersedia dalam sistem.

**Note:**

1. Transaksi payment telah ada.
2. Area Stok barang & pencatatan 🡪 kesulitan
3. Fokus produk add masa berlakuk ( FIFO)

## Audien yang Dituju dan Pembaca yang Disarankan

Dokumen ini ditujukan untuk:

* Tim Pengembang: Untuk memahami kebutuhan teknis yang harus diimplementasikan.
* Pemilik Warkop Bunny: Untuk memverifikasi bahwa sistem telah sesuai dengan proses bisnis.
* Staf Kasir/Admin: Untuk memahami alur kerja sistem yang akan mereka gunakan sehari-hari.

## Batasan Produk

* Platform hanya Android.
* Koneksi internet diperlukan karena Firebase bersifat real-time online.
* Tidak tersedia pembayaran digital dan cetak struk pada fase ini.

## Definisi dan Istilah

* SRS : *Software Requirements Specification*, atau

Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

* IEEE : *Institute of Electrical and Electronics Engineering*

Standar internasional untuk pengembangan dan perancangan produk.

* Kasir: Pengguna yang mencatat transaksi harian.
* Admin: Pengguna yang mengatur menu, stok, dan melihat laporan.
* Stok: Ketersediaan menu yang tercatat dalam sistem.
* Firebase: Platform backend yang menyediakan database real-time dan autentikasi.
* Flutter: Framework UI untuk pembuatan aplikasi lintas platform.

## Refrensi

<Tulis daftar dokumen lain atau alamat web yang menjadi acuan SRS ini.>

# Deskripsi Keseluruhan

## Deskripsi Produk

Warkop Bunny adalah sebuah warung kopi modern yang menawarkan berbagai menu makanan dan minuman, seperti Indomie berbagai topping, camilan kekinian, serta kopi instan dan minuman dingin. Aplikasi ini bertujuan menggantikan proses manual dengan sistem digital yang mampu mencatat pesanan pelanggan secara otomatis serta memantau stok bahan/menu yang tersedia.

## Fungsi Produk

Fitur utama dari aplikasi ini adalah:

* Menampilkan daftar menu berdasarkan kategori (makanan, minuman, camilan).
* Mencatat **penjualan** dan menghitung total secara otomatis.
* Mengelola stok menu (pengurangan otomatis saat transaksi dan penambahan manual oleh admin).
* Menampilkan laporan penjualan harian.
* Mengelola pengguna (admin dan kasir).

## Penggolongan Karakterik Pengguna

Terdapat dua tipe pengguna utama:

* Admin: Mengelola data stok dan menu, serta melihat laporan penjualan.
* Kasir: Mencatat transaksi penjualan harian tanpa mengubah data menu/stok.

Tabel 1 Karakteristik Pengguna

| **Kategori Pengguna** | **Tugas** | **Hak Akses ke aplikasi** | **Kemampuan yang harus dimiliki** |
| --- | --- | --- | --- |
| Admin | Manajemen menu & stok | CRUD data, laporan | Basic smartphone & admin app usage |
| Kasir | Input transaksi harian | Tambah transaksi | Entry data & pemakaian aplikasi kasir |
|  |  |  |  |
|  |  |  |  |

## Lingkungan Operasi

* Platform: Android
* Backend: Firebase Firestore DB & Auth
* Perangkat: Android smartphone minimal RAM 3GB

## Batasan Desain dan Implementasi

* Hanya mendukung Bahasa Indonesia.
* Desain antarmuka disesuaikan untuk pengguna non-teknis.
* Semua data disimpan di Firebase, tidak ada penyimpanan lokal.
* Tidak tersedia dukungan pencetakan atau mode offline.

## Dokumentasi Pengguna

<Daftar komponen dokumentasi pengguna (seperti user manual, on-line help, dan tutorial) yang akan disampaikan bersama dengan perangkat lunak yang akan dikirim>

# Kebutuhan Antarmuka Eksternal

## User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

## Hardware Interface

*<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>*

## Software Interface

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

## Communication Interface

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

# Functional Requirement

<Area ini menggambarkan pengorganisasian persyaratan fungsional untuk produk dengan fitur sistem, layanan utama yang disediakan oleh produk>

<Tulis Kebutuhan Fungsional / Functional Requirement disini>

Diawali dengan membuat daftar kebutuhan fungsional P/L, lengkap dengan ID dan penjelasan jika perlu. Bisa dibuat dalam bentuk tabel.

| **ID** | **Kebutuhan Fungsional** | **Penjelasan** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Use Case Diagram

<*Gambarkan use case diagramnya dari functional requirement yang didapatkan*>

## Nama Use Case 1

4.1.1 Deskripsi Use Case

<desripsikan / jabarkan mengenai use case ini >

4.1.2 Stimulus and Respon

<menyediakan daftar aksi yang dilakukan oleh user dan respon dari sistem.>

|  |  |
| --- | --- |
| Action by user | Response from system |
| 1 |  |
|  | 2 |
| 3 |  |
|  | 4 .. |

4.1.4 *Activity Diagram*

## Nama Use Case 2

<Sama seperti di atas, dan seterusnya sesuai jumlah use case yang didapatkan>

## Class Diagram

<*identifikasi kelas yang terkait dan hubungannya pada sistem yang dikembangkan*>

# Non Functional Requirements

<*Uraikan dengan ringkas kebutuhan non fungsional dalam tabel sebagai berikut. Isilah Kolom Kebutuhan dengan kalimat yang jelas dan kelak dapat ditest untuk dipenuhi. ID adalah nomor kebutuhan yang harus ditelusuri pada saat test. Tuliskan N/A bila Not Applicable>*

| **ID** | **Parameter** | **Kebutuhan** |
| --- | --- | --- |
|  | Availability |  |
|  | Reliability |  |
|  | Ergonomy |  |
|  | Portability |  |
|  | Memory |  |
|  | Response time |  |
|  | Safety | N/A |
|  | Security |  |
|  |  |  |
|  | Others 1: Bahasa komunikasi | Misalnya : semua tanya jawab harus dalam bahasa Indonesia |
|  |  | Setiap layar harus mengandung logo PT Pos Indonesia |
|  |  |  |

Catatan :

*Availability : ketersediaan aplikasi, misalnya harus terus menerus beroperasi 7 hari perminggu, 24 jam per haritanpa gagal*

*Reliability : keandalan, misalnya tidak pernah boleh gagal(atau kegagalan yang ditolerir adalah …%) sehingga harus dipikirkan fault tolerant architecture. Biasanya hanya perlu untuk Critical Application yang jika gagal akan berakibat fatal.*

*Ergonomy : kenyamanan pakai bagi pengguna*

*Portability : kemudahan untuk dibawa dan dioperasikan ke mesin/sistem operasi/platform yang lain*

*Memory : jika perhitungan kapasitas memori internal kritis (misalnya untuk SW yang harus dijadikan CHIPS dan ukurannya harus kecil*

*Response time : Batasan waktu yang harus dipenuhi. Sangat penting untuk aplikasi Real Time. Contoh: “Aaplikasi harus mampu menampilkan hasil dalam 4 detik”, atau “ATM harus menarik kembali kartu yang tidak diambil dalam waktu 3 menit”*

*Safety: yang menyangkut keselamatan manusia, misalnya untuk SW yang dipakai pada sistem kontrol di pabrik*

*Security : aspek keamanan yang harus dipenuhi*