

**BUSINESS REQUIREMENTS DOCUMENT (BRD)
SOFTWARE REQUIREMENTS SPECIFICATION (SRS)**



Dosen Pengampu:

Jefry Sunupurwa, S.Kom., M.Kom

Disusun Oleh:

Azizaturridha Fw – 20230801090

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS ESA UNGGUL TANGERANG**

BUSINESS REQUIREMENT DOCUMENT (BRD)

Toko Kue Tugas Kuliah

1. Ringkasan Bisnis

Nama Proyek: Sistem Manajemen Toko Roti Online

Deskripsi Singkat:

Sistem manajemen toko roti berbasis web ini akan memungkinkan pelanggan untuk membeli berbagai produk roti dengan mudah. Dengan sistem ini, pelanggan dapat melihat katalog produk, menambahkan produk ke keranjang belanja, memilih metode pembayaran, dan melakukan pemesanan secara langsung melalui platform online.

Latar Belakang Bisnis:

Toko roti sering kali memiliki tantangan dalam hal pengelolaan penjualan, stok produk, dan manajemen pesanan. Dengan semakin berkembangnya teknologi, banyak konsumen yang lebih memilih berbelanja secara online, termasuk membeli roti. Oleh karena itu, diperlukan sebuah sistem yang memungkinkan pelanggan untuk memesan roti secara online, mengelola pesanan, dan mempercepat proses transaksi tanpa harus datang ke toko.

Selain itu, proyek ini menggunakan **Laravel boilerplate** untuk mempercepat proses pengembangan dengan struktur yang sudah rapi dan best practice Laravel, sehingga pengembangan menjadi lebih efisien dan mudah dipelihara. API yang dihasilkan akan terdokumentasi menggunakan **L5-Swagger**, memudahkan integrasi dengan aplikasi lain atau layanan mobile. Antarmuka aplikasi dibangun dengan **Bootstrap template**, sehingga tampilan UI responsif, modern, dan mudah digunakan.

A. Informasi Umum

Saat ini, toko roti kami hanya menerima pesanan secara manual di toko fisik. Dengan adanya aplikasi berbasis web, kami ingin memberikan kenyamanan lebih bagi pelanggan yang ingin memesan roti secara online. Sistem ini akan terintegrasi dengan sistem pembayaran digital dan juga memudahkan admin toko dalam mengelola pesanan dan stok.

2. Executive Summary

Sistem manajemen toko roti ini akan mengintegrasikan berbagai fitur untuk mempermudah pelanggan dalam melakukan pembelian produk roti secara online. Sistem ini juga akan memungkinkan pengelolaan produk, stok, dan pesanan oleh pihak admin, serta memberikan laporan yang akurat untuk analisis penjualan.

3. Objective Summary

Tujuan utama dari sistem ini adalah untuk:

1. Mempermudah pelanggan dalam memilih, membeli, dan memesan roti secara online.
2. Menyediakan sistem backend yang efisien untuk pengelolaan produk, pesanan, dan laporan penjualan.
3. Memberikan pengalaman pengguna yang baik dengan antarmuka yang responsif dan mudah digunakan.

4. Goals

Goals Sistem Manajemen Tugas dan Proyek

1. Membangun Sistem CRUD untuk Tasks dan Projects

- Mengembangkan **fitur CRUD** (Create, Read, Update, Delete) untuk **Tasks (Tugas Teori)** dan **Projects (Tugas Besar)** dalam sistem manajemen.
- Memastikan pengguna dapat membuat, mengelola, dan menghapus tugas teori dan proyek secara efisien dengan antarmuka yang intuitif.

2. Menyediakan Admin Panel dengan Filament

- **Admin Panel** yang dibangun menggunakan **Filament** untuk memberikan kemudahan dalam manajemen data.
- Memungkinkan admin untuk melakukan **manajemen pengguna, pengelolaan tugas, pengelolaan proyek**, serta **melihat laporan dan statistik** dengan antarmuka yang bersih dan user-friendly.

3. Menyediakan API Lengkap dengan Dokumentasi Otomatis (L5-Swagger)

- Mengembangkan **API** yang dapat diakses untuk berinteraksi dengan sistem, termasuk endpoint untuk menambah, mengubah, dan menghapus tugas serta proyek.
- Menyediakan **dokumentasi otomatis** untuk API menggunakan **L5-Swagger**, mempermudah pengembang dalam mengintegrasikan aplikasi lain atau pengujian API.

4. Memberikan UI yang Rapi, Responsif, dan Mudah Digunakan

- Membangun **User Interface (UI)** yang responsif menggunakan **Bootstrap**, memastikan tampilan yang bersih, konsisten, dan mudah dinavigasi di perangkat desktop maupun mobile.
- Menyediakan **pengalaman pengguna (UX)** yang optimal dengan navigasi yang sederhana dan fungsionalitas yang mudah dimengerti.

5. Memastikan Struktur Kode Rapi dan Scalable melalui Laravel Boilerplate

- Menggunakan **Laravel Boilerplate** untuk memastikan **struktur kode** yang **bersih, terorganisir, dan scalable**.
- Memastikan aplikasi dapat dengan mudah ditingkatkan atau dimodifikasi di masa depan dengan mengikuti **best practices** Laravel untuk pengembangan aplikasi yang mudah dipelihara.

5. Cara Mewujudkan

Teknologi yang digunakan:

- Laravel 12 dengan boilerplate untuk kerangka kerja standar.
- Filament v3 sebagai admin panel CRU
- L5-Swagger untuk dokumentasi API otomatis.
- MySQL untuk basis data.

Langkah Pengembangan:

1. Rancang database untuk produk roti, kategori, dan pesanan.
2. Buat CRUD untuk produk roti (Menambah, mengedit, menghapus).
3. Integrasikan sistem pembayaran online.
4. Implementasikan sistem pemesanan yang memungkinkan pelanggan memilih produk dan melakukan checkout.
5. Bangun halaman admin untuk mengelola pesanan dan laporan.

6. Role-Based Access Control (RBAC)

1. Admin:

- Mengelola produk roti, harga, dan stok.
- Memproses pesanan dan mengonfirmasi pembayaran.
- Menyediakan laporan penjualan.

2. Pelanggan:

- Melakukan pencarian produk roti.
- Menambahkan produk ke keranjang dan melakukan pembayaran.
- Melihat status pesanan.

7. Scope of Work

Fitur yang akan dikerjakan dalam lingkup proyek ini:

1. Manajemen produk roti (CRUD).
2. Sistem manajemen pesanan.

3. Integrasi pembayaran online.
4. Dashboard admin untuk pengelolaan produk dan laporan penjualan.
5. Halaman frontend untuk pencarian produk, pemesanan, dan pembayaran.

Fitur yang akan dikerjakan dalam lingkup proyek ini:

- Desain database untuk tasks dan projects.
- Implementasi CRUD lengkap untuk Tasks dan Projects.
- Integrasi panel admin Filament untuk manajemen data.
- Pembuatan API endpoint untuk Tasks dan Projects.
- Dokumentasi API menggunakan L5-Swagger.
- Implementasi antarmuka Bootstrap responsif untuk tampilan frontend dasar.
- Uji coba fungsi, dokumentasi, dan performa.

8. Analisis Kebutuhan

A. Kebutuhan Fungsional

- CRUD Tasks: Admin dapat menambah, mengedit, menghapus, dan melihat data tugas teori (Tasks).
- Admin dapat menandai status tugas selesai atau belum.
- CRUD Projects: Admin dapat menambah, mengedit, menghapus, dan melihat data tugas besar atau praktikum (Projects).
- Admin dapat menentukan deadline dan deskripsi untuk setiap project.
- Admin Panel Filament: Menyediakan antarmuka manajemen data yang ramah pengguna dan cepat diakses.
- API dengan Dokumentasi L5-Swagger: Semua endpoint Tasks dan Projects terdokumentasi otomatis serta memungkinkan integrasi dengan aplikasi atau layanan lain di masa depan.
- UI Responsif: Halaman frontend sederhana dengan Bootstrap, menampilkan daftar tugas dan proyek dengan tampilan modern.

B. Kebutuhan Non-Fungsional

Kebutuhan Non-Fungsional untuk Sistem Toko Kue Online

1. Performa dan Skalabilitas:

- Sistem harus mampu menangani jumlah data produk dan pesanan yang terus meningkat tanpa penurunan performa.
- Sistem dapat menangani lonjakan trafik pada waktu-waktu tertentu, misalnya saat promo atau event tertentu, tanpa mengalami keterlambatan atau crash.

2. Kemudahan Pemeliharaan:

- Dengan menggunakan **boilerplate Laravel**, struktur kode yang rapi dan standar memudahkan tim pengembangan lain untuk memahami dan memelihara sistem.

- Sistem dibuat modular agar mudah untuk menambahkan atau mengubah fitur tanpa mengganggu bagian lain dari aplikasi.

3. **Antarmuka Responsif:**

- Antarmuka pengguna harus menggunakan **Bootstrap** atau framework CSS lainnya untuk memastikan tampilan yang adaptif dan responsif di berbagai ukuran layar, termasuk desktop, tablet, dan smartphone.
- Halaman produk dan pemesanan harus mudah dinavigasi, dan elemen-elemen UI harus menyesuaikan diri dengan baik di berbagai perangkat.

4. **Keamanan Dasar:**

- Hanya **admin** yang memiliki akses ke **panel manajemen data**, memastikan keamanan data pelanggan dan pesanan.
- API hanya digunakan untuk **integrasi yang sah**, dan hanya pihak yang berwenang yang dapat mengakses dan melakukan operasi pada data pesanan dan produk.
- Data pelanggan dan transaksi harus dienkripsi dan dilindungi dengan baik menggunakan HTTPS dan sistem otentikasi yang kuat.

5. **Ketersediaan Dokumentasi:**

- **Dokumentasi API** harus selalu tersedia dan up-to-date melalui **L5-Swagger**, sehingga pengembang atau pihak ketiga yang membutuhkan integrasi dapat dengan mudah memahami dan menggunakan API yang disediakan.
- Setiap endpoint API akan didokumentasikan secara otomatis dan akan mencakup deskripsi, parameter yang dibutuhkan, serta contoh penggunaan.

9. Stakeholder

Stakeholder	Peran	Kebutuhan / Tanggung Jawab
Pelanggan	Pengguna yang membeli produk kue online	- Memilih produk kue - Melakukan pemesanan dan pembayaran
Admin Toko	Pengelola toko yang mengelola produk dan pesanan	- Mengelola produk dan pesanan - Memastikan ketersediaan stok
Pengembang Sistem	Tim teknis yang membangun dan memelihara sistem	- Membangun aplikasi yang aman dan responsif - Memelihara sistem

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

Aplikasi Manajemen Toko Kue Online

1.1 Tujuan

Dokumen ini bertujuan untuk memberikan spesifikasi lengkap mengenai **Aplikasi Manajemen Toko Kue Online**. Sistem ini membantu pelanggan untuk membeli kue secara online dengan mudah, serta memungkinkan admin untuk mengelola produk kue, pesanan, dan transaksi dengan efisien.

1.2 Ruang Lingkup Sistem

Aplikasi **Toko Kue Online** dibangun menggunakan **Laravel 12** dengan boilerplate untuk struktur yang konsisten. Sistem menyediakan fitur CRUD untuk entitas **Produk Kue** dan **Pesanan** melalui panel admin **Filament**. API terdokumentasi menggunakan **L5-Swagger**, dan antarmuka dirancang responsif menggunakan **Bootstrap**.

1.3 Akronim dan Definisi

- **CRUD**: Create, Read, Update, Delete.
- **Produk Kue**: Produk kue yang dijual di toko roti online.
- **Pesanan**: Transaksi pemesanan produk kue oleh pelanggan.
- **Bootstrap**: Framework CSS untuk membuat antarmuka web responsif.
- **L5-Swagger**: Paket Laravel untuk dokumentasi API otomatis.
- **Boilerplate Laravel**: Template dasar Laravel dengan struktur siap pakai.

2. Deskripsi Umum

2.1 Perspektif Produk

Produk ini adalah aplikasi mandiri berbasis web yang dirancang untuk digunakan oleh pelanggan dan admin toko kue. Aplikasi dibangun di atas **Laravel boilerplate** dengan struktur **MVC** yang jelas. Panel admin menggunakan **Filament** untuk manajemen data, sedangkan API disediakan dan terdokumentasi melalui **L5-Swagger**.

2.2 Fungsi Utama

- Mengelola daftar **Produk Kue** (termasuk nama, deskripsi, harga, gambar, dan kategori).
- Mengelola daftar **Pesanan** (termasuk data pelanggan, produk yang dipesan, jumlah, alamat pengiriman, metode pembayaran, dan status pesanan).
- Menampilkan ringkasan data produk dan pesanan pada panel admin.
- Menyediakan API untuk integrasi eksternal dengan dokumentasi otomatis menggunakan **L5-Swagger**.

- Menyediakan antarmuka web berbasis **Bootstrap** yang mudah digunakan.

2.3 Karakteristik Pengguna

Pengguna utama adalah **admin toko** yang bertugas mengelola produk dan pesanan, serta **pelanggan** yang membeli kue secara online. Admin memiliki akses penuh untuk menambah, mengedit, menghapus, dan melihat data produk dan pesanan. Pelanggan diharapkan dapat dengan mudah melakukan pemesanan melalui antarmuka web yang responsif.

12. Kebutuhan Fungsional

- Admin dapat **menambah, mengedit, menghapus, dan melihat data Produk Kue**.
- Admin dapat **menambah, mengedit, menghapus, dan melihat data Pesanan**.
- Panel admin menampilkan daftar produk kue dan pesanan dengan fitur pencarian dan filter dasar.
- API menyediakan **endpoint CRUD** untuk **Produk Kue** dan **Pesanan**, lengkap dengan dokumentasi **L5-Swagger**.

13. Kebutuhan Non-Fungsional

- **Responsif**: Sistem menggunakan **Bootstrap** untuk memastikan tampilan dapat berfungsi dengan baik di perangkat desktop, tablet, dan smartphone.
- **Kemudahan Pemeliharaan**: Dengan **boilerplate Laravel**, struktur kode tetap rapi dan konsisten, memudahkan pengembangan dan pemeliharaan sistem di masa depan.
- **Keamanan**: Akses admin terbatas untuk mengelola produk dan pesanan, serta transaksi pelanggan dienkripsi menggunakan **HTTPS**.
- **Dokumentasi API**: API selalu terdokumentasi dengan baik dan selalu up-to-date menggunakan **L5-Swagger**, memungkinkan integrasi dengan aplikasi lain.

14. Desain Basis Data

Tabel: produk_kue

Kolom	Tipe Data	Keterangan
id	bigint (auto increment)	Primary key
name	varchar	Nama produk kue
description	text	Deskripsi produk
price	decimal	Harga produk
image	varchar	Path gambar produk
created_at	timestamp	Tanggal dibuat
updated_at	timestamp	Tanggal diperbarui

Tabel: pesanan

Kolom	Tipe Data	Keterangan
id	bigint (auto increment)	Primary key
user_id	bigint	ID pelanggan (Foreign Key)
total_price	decimal	Total harga pesanan
status	varchar	Status pesanan (pending, selesai)
created_at	timestamp	Tanggal dibuat
updated_at	timestamp	Tanggal diperbarui

15. Arsitektur Sistem

Aplikasi menggunakan arsitektur **Model-View-Controller (MVC)**:

- **Model: Produk Kue dan Pesanan** berinteraksi dengan database.
- **View: Blade template** dengan **Bootstrap** untuk menampilkan data di frontend.
- **Controller:** Mengatur alur data antara model dan view.
- **Filament:** Digunakan untuk panel admin yang memungkinkan CRUD produk dan pesanan.
- **L5-Swagger:** Digunakan untuk mendokumentasikan API yang tersedia.

Server menggunakan **Nginx** dengan **PHP 8.2** di atas sistem operasi **Ubuntu**. Database menggunakan **MySQL**.

16. Pengembangan Masa Depan

- Penambahan **fitur autentikasi pelanggan** dan **admin** dengan role berbeda (misalnya pelanggan biasa dan admin).
- Penambahan **notifikasi otomatis** melalui email atau WhatsApp untuk pelanggan terkait status pesanan atau promosi.
- **Integrasi dengan aplikasi mobile** untuk memungkinkan pelanggan melakukan pemesanan melalui smartphone.
- Penambahan fitur **loyalty program** untuk pelanggan setia, termasuk diskon dan poin reward.

