

LAPORAN ANALISIS DAN PERANCANGAN SISTEM RAY ENTERTAINMENTS (RANTS)

Fokus Utama: Digitalisasi Layanan Sanggar Tari dengan Metode Extreme Programming (XP) dan Desain Modern Profesional.

BAB 1: PENDAHULUAN

1.1 Latar Belakang Masalah

Sistem pengelolaan layanan Sanggar Tari Ray Entertainments (RANTS) saat ini masih dilakukan secara manual dan terfragmentasi (melalui WhatsApp dan media sosial). Metode ini menyebabkan inefisiensi administrasi, sulitnya pengelolaan stok kostum, dan sering terjadinya miskomunikasi atau bentrok jadwal pemesanan jasa tari/make up.

1.2 Tujuan Proyek

- 1. Menciptakan sistem *multiplatform* (Aplikasi Mobile Customer dan Website Admin) yang terintegrasi.
- 2. Menerapkan metodologi **Extreme Programming (XP)** untuk memastikan pengembangan yang cepat, adaptif, dan responsif terhadap *feedback* pengguna.
- 3. Menyediakan sistem yang meningkatkan transparansi informasi, keteraturan administrasi, dan pengalaman pengguna (*user experience*).

BAB 2: ANALISIS KEBUTUHAN SISTEM

Analisis Kebutuhan telah mengidentifikasi tiga jenis layanan utama dan kebutuhan fungsional yang berbeda untuk Customer dan Admin.

2.1 Kebutuhan Fungsional (Aplikasi Mobile - Customer)

N o.	Fitur	Kebutuhan Detail
1	Katalog & Pemesanan	Menampilkan Kostum (dengan stok), Jasa Tari (dengan paket), dan Make Up (dengan kategori harga) secara visual dan informatif.
2	Penjadwalan	Memungkinkan pemilihan tanggal mulai dan selesai untuk <i>booking</i> layanan, dengan validasi bentrok jadwal.
3	Pembayaran PG	Integrasi penuh dengan Payment Gateway (e.g., Midtrans/Xendit) untuk pembayaran VA dan <i>E-Wallet</i> .
4	Riwayat & Ulasan	Melihat <i>history</i> order, status terkini, dan memberikan Rating (1-5) serta Ulasan.

2.2 Kebutuhan Fungsional (Website - Admin)

No	Fitur	Kebutuhan Detail
.		
1	CRUD Layanan Spesifik	Mengelola data di tabel <code>costumes</code> , <code>dance_services</code> , dan <code>makeup_services</code> secara terpisah dan terperinci.
2	Validasi Stok/Jadwal	Mengelola <code>stock_log</code> untuk Kostum dan memantau ketersediaan slot tanggal secara visual (Dashboard).
3	Manajemen Pemesanan	Mengubah status pesanan (<code>orders</code>) dan memproses konfirmasi pembayaran.
4	Pelaporan Transaksi	Melihat data <code>transactions</code> dan merekonsiliasi pembayaran dari PG.

BAB 3: RANCANGAN ARSITEKTUR DAN DATABASE

3.1 Arsitektur Sistem

Menggunakan arsitektur *Multi-tier* dengan:

- Backend (Business Logic & Data Layer): Framework Laravel (PHP)
- Database (Persistence Layer): MySQL (dengan skema relasional terstruktur)
- Frontend Mobile: Flutter (Dart)
- Frontend Web Admin: Laravel Blade + Tailwind CSS (Fokus pada Desain Modern)

3.2 Rancangan Tabel Database (Final)

Struktur tabel database telah disempurnakan untuk memastikan akurasi stok *real-time* dan manajemen jadwal. Berikut adalah rincian lengkap setiap tabel:

3.2.1 Tabel: `users`

Fungsi Utama: Menyimpan data pengguna (Customer aplikasi) dan pengelola (Admin website) untuk keperluan autentikasi dan otorisasi.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
<code>id</code>	BIGINT (PK)	Primary Key (Kunci Utama).
<code>name</code>	VARCHAR(255)	Nama lengkap pengguna.
<code>email</code>	VARCHAR(255)	Alamat email (Username untuk login). UNIQUE .
<code>phone_number</code>	VARCHAR(15)	Nomor telepon pengguna.
<code>password</code>	VARCHAR(255)	Kata sandi yang di- <i>hash</i> .

role	ENUM('customer', 'admin')	Peran pengguna. Menentukan akses ke Aplikasi Mobile atau Website Admin.
created_at	TIMESTAMP	Waktu pembuatan data.

3.2.2 Tabel: costumes

Fungsi Utama: Menyimpan inventaris lengkap untuk layanan Sewa Kostum, termasuk harga dan stok fisik.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
costume_name	VARCHAR(255)	Nama spesifik kostum (e.g., "Tari Melayu Kuning").
description	TEXT	Detail baju, warna, bahan, aksesoris.
rental_price	DECIMAL(10, 2)	Harga sewa per periode.
stock	INT	Total stok fisik yang dimiliki.
size	VARCHAR(50)	Ukuran (e.g., 'All Size', 'M', 'L').
image_path	VARCHAR(255)	Path file gambar kostum.

3.2.3 Tabel: dance_services

Fungsi Utama: Menyimpan daftar paket harga untuk layanan Jasa Tari berdasarkan jumlah penari.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
min_dancers	INT	Jumlah minimal penari dalam paket (Wajib Ganjil, e.g., 3, 5, 7).
price	DECIMAL(10, 2)	Harga untuk paket jumlah penari ini.
description	TEXT	Keterangan paket (jenis tari, durasi).

3.2.4 Tabel: makeup_services

Fungsi Utama: Menyimpan daftar harga untuk layanan Jasa Make Up berdasarkan kategori pengguna/acara.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.

category	ENUM('SD', 'SMP', 'SMA', 'Wisuda', 'Acara Umum')	Kategori penentuan harga.
price	DECIMAL(10, 2)	Harga sesuai kategori.
description	TEXT	Detail layanan make up per kategori.

3.2.5 Tabel: orders

Fungsi Utama: Menyimpan data induk dari setiap transaksi pemesanan layanan dan informasi penjadwalan.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
user_id	BIGINT (FK)	Foreign Key ke users . Siapa yang memesan.
start_date	DATE	Tanggal mulai pemakaian/pelaksanaan jasa.
end_date	DATE	Tanggal akhir pemakaian/pengembalian (penting untuk kostum).
total_amount	DECIMAL(10, 2)	Total biaya dari semua item dalam pesanan.
status	ENUM(...)	Status pesanan: pending , paid , confirmed , completed , cancelled .
return_status	ENUM('belum', 'terlambat', 'sudah')	Status pengembalian kostum.
created_at	TIMESTAMP	Waktu pesanan dibuat.

3.2.6 Tabel: order_details

Fungsi Utama: Menghubungkan pesanan (orders) ke item layanan spesifik dari tabel detail.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
order_id	BIGINT (FK)	Foreign Key ke orders .
service_type	ENUM('kostum', 'tari', 'rias')	Jenis layanan yang dipesan (Menentukan tabel detail mana yang diacu).

detail_id	BIGINT	ID dari tabel detail yang relevan (costumes.id , dance_services.id , atau makeup_services.id).
quantity	INT	Jumlah unit yang dipesan.
unit_price	DECIMAL(10, 2)	Harga satuan layanan saat pemesanan.

3.2.7 Tabel: transactions

Fungsi Utama: Mencatat detail pembayaran dan krusial untuk integrasi Payment Gateway (PG).

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
order_id	BIGINT (FK)	Foreign Key ke orders . UNIQUE (1 order = 1 transaksi).
transaction_code	VARCHAR(255)	ID Transaksi unik dari Payment Gateway (PG).
payment_method	VARCHAR(50)	Metode pembayaran (e.g., 'BCA VA', 'DANA', 'GoPay').
va_number	VARCHAR(50)	Nomor Virtual Account (Jika berlaku).
amount	DECIMAL(10, 2)	Jumlah yang dibayarkan.
pg_status	VARCHAR(50)	Status resmi dari PG (e.g., 'settlement', 'pending', 'expire').
paid_at	TIMESTAMP	Waktu pembayaran berhasil (diterima dari Webhook PG).

3.2.8 Tabel: stock_log

Fungsi Utama: Mencatat setiap pergerakan stok Kostum dan menjadi sumber data utama untuk validasi stok pada tanggal tertentu (Anti *Double Booking*).

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
costume_id	BIGINT (FK)	Foreign Key ke costumes .
order_id	BIGINT (FK)	Foreign Key ke orders .
quantity_change	INT	Jumlah perubahan stok: Negatif (saat disewa), Positif (saat dikembalikan).
log_date	DATE	Tanggal kejadian (Penting untuk validasi jadwal).

type	ENUM('out', 'in', 'adjustment')	Tipe log.
------	---------------------------------	-----------

3.2.9 Tabel: reviews

Fungsi Utama: Menyimpan data ulasan dan rating yang diberikan oleh Customer.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
order_id	BIGINT (FK)	Foreign Key ke orders .
user_id	BIGINT (FK)	Foreign Key ke users .
rating	TINYINT	Nilai rating (1-5).
comment	TEXT	Ulasan/komentar dari pengguna.

3.2.10 Tabel: gallery

Fungsi Utama: Menyimpan path gambar untuk Galeri Sanggar yang dikelola oleh Admin.

Kolom	Tipe Data (Simulasi)	Keterangan & Constraint
id	BIGINT (PK)	Primary Key.
title	VARCHAR(255)	Judul foto.
image_path	VARCHAR(255)	Lokasi penyimpanan file gambar.
uploaded_by	BIGINT (FK)	ID Admin yang mengunggah (FK ke users).

BAB 4: ALUR BISNIS DAN INTERAKSI KRITIS

4.1 Alur Pengurangan Stok Kostum (Anti Double Booking)

1. **Validasi Ketersediaan:** Saat *Customer* memilih tanggal (*start_date* ke *end_date*), sistem menghitung *stok available* dengan menjumlahkan *quantity_change* dari tabel *stock_log* dalam rentang tanggal tersebut.
2. **Order Dibuat:** Jika stok cukup, *order* dibuat (*orders.status* = 'pending').
3. **Log Out:** Setelah pembayaran dikonfirmasi (*orders.status* = 'paid'), sistem mencatat **pengurangan stok** ke tabel *stock_log* dengan *quantity_change* bernilai negatif (-N) dan *log_date* adalah *start_date* dan *end_date*.
4. **Log In:** Setelah Admin mengkonfirmasi pengembalian (*orders.return_status* = 'sudah'), sistem mencatat **penambahan stok** dengan *quantity_change* bernilai positif (+N) dan *log_date* adalah tanggal pengembalian.

4.2 Integrasi Payment Gateway (PG)

Sistem Laravel harus memiliki *endpoint Webhook Listener* yang aman. *Endpoint* ini menerima notifikasi *real-time* dari PG (Midtrans/Xendit) untuk memperbarui status transaksi tanpa campur tangan manual.

BAB 5: FOKUS DESAIN WEB ADMIN (MODERN & PROFESIONAL)

Desain *Website Admin* harus mencerminkan profesionalisme dan efisiensi, mengingat ini adalah alat kerja utama *Owner*.

Aspek Desain	Kriteria Modern & Profesional	Implementasi Teknikal
Estetika	Bersih, Minimalis, dengan palet warna yang konsisten (Maroon/Merah Bata RANTS sebagai aksen).	Menggunakan Tailwind CSS untuk konsistensi dan adaptabilitas.
Tipografi	Penggunaan font Sans-Serif yang mudah dibaca (<i>Inter</i> atau sejenisnya) dengan hierarki yang jelas.	Kelas Tailwind untuk font dan ukuran teks yang terstruktur.
Dashboard	Tampilan utama yang menyajikan ringkasan data penting (<i>Total Order Pending, Stok Kritis, Jadwal Hari Ini</i>) dalam bentuk kartu dan <i>chart</i> sederhana.	Menggunakan Grid Layout dan komponen Card.
Navigasi	Navigasi Samping (Sidebar) yang konsisten dan jelas, dengan ikon visual untuk setiap menu (Dashboard, Layanan, Pemesanan, Laporan).	Sidebar <i>sticky</i> dengan desain minimalis dan responsif untuk layar kecil.