



Jurnal SANTI (Sistem Informasi dan Teknologi Informasi)

Vol. 5 No.2 Tahun 2026

DOI: <https://doi.org/10.58794/santi.v5i2.2270>

Perancangan Aplikasi *E-commerce* Berbasis Android pada UMKM Ranitha Stuff Menggunakan Metode Prototype

Naia Tantia Putri¹, Geardo Lapista Simamora², Sherly Nadia Refica³, Fatiah Azzahra Gea⁴, Raihanah⁵, Angga Pratama^{6*}

^{1,2,3,4,5,6} Sistem Informasi, Fakultas Teknik, Universitas Malikussaleh, Lhokseumawe, Indonesia

e-mail: ¹naia.230180097@mhs.unimal.ac.id; ²geardo.230180093@mhs.unimal.ac.id;

³sherly.230180095@mhs.unimal.ac.id; ⁴fatiah.230180096@mhs.unimal.ac.id;

⁵raihanah.230180098@mhs.unimal.ac.id; ^{6*}anggapratama@unimal.ac.id

* Corresponding author

(Received : 15 Juni 2026; Revised: 24 Juni 2026; Accepted: 7 Juli 2026; Available online: 30 Juli 2026)

Abstrak

Perkembangan teknologi digital mendorong pertumbuhan *e-commerce* di Indonesia, sekaligus membuka peluang bagi UMKM untuk memperluas jangkauan pasar secara digital. UMKM Ranitha Stuff yang bergerak di bidang kosmetik, aksesoris, dan perawatan kulit menghadapi permasalahan berupa penurunan penjualan saat libur semester dan belum terintegrasinya proses penjualan online dalam satu platform mandiri. Penelitian ini bertujuan merancang dan membangun aplikasi *e-commerce* berbasis Flutter pada UMKM Ranitha Stuff menggunakan metode Prototype. Sistem dikembangkan menggunakan Flutter, Laravel, MySQL, dan integrasi payment gateway Midtrans, dengan fitur utama meliputi manajemen produk, keranjang belanja, checkout, pelacakan pesanan, chat, serta rating dan ulasan produk. Pengujian dilakukan menggunakan metode Black Box Testing dan Performance Testing. Hasil Black Box Testing terhadap 10 fitur utama menunjukkan tingkat keberhasilan sebesar 100%, sedangkan Performance Testing menghasilkan rata-rata response time API sebesar 2,86 detik dan frame rate aplikasi sebesar 59 FPS. Hasil tersebut menunjukkan bahwa sistem telah memenuhi kebutuhan fungsional serta memiliki performa yang baik. Aplikasi ini diharapkan dapat membantu UMKM Ranitha Stuff dalam meningkatkan efisiensi penjualan digital dan mengurangi ketergantungan terhadap marketplace pihak ketiga.

Kata kunci: *E-commerce*, Flutter, Laravel, Midtrans, UMKM

Abstract

The development of digital technology has driven the growth of *e-commerce* in Indonesia while creating opportunities for SMEs to expand their market reach through digital platforms. Ranitha Stuff, an SME operating in the cosmetics, accessories, and skincare sector, faces challenges including declining sales during semester breaks and the absence of an integrated standalone online sales platform. This study aims to design and develop a Flutter-based *e-commerce* application for Ranitha Stuff using the Prototype method. The system was developed using Flutter, Laravel, MySQL, and the Midtrans payment gateway, with key features including product management, a shopping cart, checkout, order tracking, chat, and product ratings and reviews. The system was evaluated using Black Box Testing and Performance Testing. Black Box Testing of the 10 main features achieved a 100% success rate, while Performance Testing showed an average API response time of 2.86 seconds and an average application frame rate of 59 FPS. These results indicate that the system meets the functional requirements and delivers

stable performance. The application is expected to improve the digital sales efficiency of Ranitha Stuff and reduce its dependence on third-party marketplaces.

Keywords: *E-commerce, Flutter, Laravel, Midtrans, SMEs*

1. Pendahuluan

Dimulainya era globalisasi baru dengan kontribusi digital yang meningkat terutama pada sektor perekonomian [1] telah menyebabkan persaingan yang lebih ketat di dunia bisnis [2]. Penggunaan teknologi informasi untuk mendukung bisnis sangat beragam, salah satunya adalah perdagangan elektronik atau *e-commerce* [3]. *E-commerce* berperan dalam mendukung pertumbuhan bisnis dengan memperluas jangkauan pasar serta memungkinkan pelaku usaha menjangkau konsumen secara lebih luas melalui platform digital [4].

Usaha Mikro, Kecil, dan Menengah (UMKM) merupakan elemen penting dalam perekonomian nasional yang berperan besar dalam membentuk kebijakan dan struktur ekonomi strategis. Jumlah Usaha Mikro, Kecil dan Menengah (UMKM) di Indonesia sekitar 62 juta entitas selama periode 2017 hingga 2018 [5]. Pada tahun 2018, sektor UMKM berhasil mempekerjakan 97,0% dari keseluruhan angkatan kerja Indonesia dan menyumbang 61,1% terhadap Produk Domestik Bruto (PDB) nasional [6]. Kemajuan UMKM sangat penting tidak semata-mata untuk memperkuat ekspansi ekonomi, tetapi juga untuk menghasilkan peluang kerja, mengurangi kemiskinan, dan mempromosikan inovasi di berbagai wilayah [5].

Pertumbuhan *e-commerce* di Indonesia terus menunjukkan tren yang signifikan. Berdasarkan publikasi Statistik *E-Commerce* 2024 [7], jumlah usaha *e-commerce* pada tahun 2024 tercatat sebanyak 4,40 juta usaha, meningkat 15,3% dibandingkan tahun sebelumnya. Selain itu, media penjualan *online* yang paling banyak digunakan adalah pesan instan seperti WhatsApp (94,76%), diikuti media sosial (29,66%), dan *marketplace* (17,23%). Data tersebut menunjukkan bahwa sebagian besar pelaku usaha masih belum memanfaatkan platform *e-commerce* mandiri secara optimal, sehingga masih terdapat peluang besar bagi pengembangan sistem penjualan digital yang lebih terintegrasi.

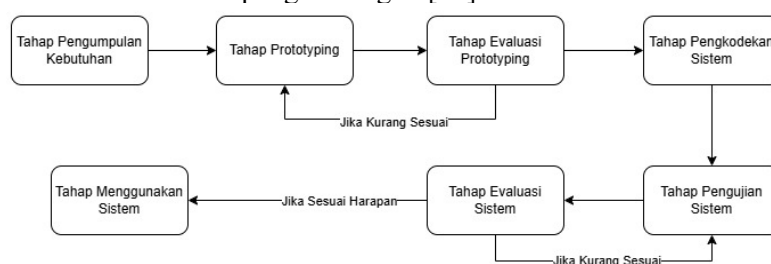
Dalam meningkatkan penjualan dan pemasaran, diperlukan sistem digitalisasi yang dapat membantu pelaku usaha mempromosikan produknya dengan lebih mudah [8]. Saat ini, banyak pelaku usaha memanfaatkan platform *e-commerce* sebagai sarana untuk memasarkan produk maupun layanan secara digital [2]. Agar proses digitalisasi pada UMKM dapat berjalan secara optimal, diperlukan kesiapan UMKM untuk bertransisi ke sistem penjualan digital melalui *e-commerce* [8]. Salah satu teknologi yang terus berkembang untuk mendukung proses tersebut adalah teknologi *mobile* berbasis Android [9]. Aplikasi *mobile* telah menjadi bagian penting dalam ekosistem bisnis modern [10], sementara platform Android menjadi salah satu sistem yang paling banyak digunakan karena mampu mempermudah akses informasi bagi pengguna [9].

Ranitha Stuff adalah usaha kecil menengah yang bergerak di bidang kosmetik, aksesoris, dan perawatan kulit. Tujuan utama penjualan usaha ini adalah menjual produk kepada mahasiswa serta masyarakat sekitar, karena lokasi toko terletak dekat area kampus. Namun, berdasarkan hasil wawancara yang dilakukan, penjualan di Ranitha Stuff mengalami penurunan selama masa libur semester karena aktivitas mahasiswa di sekitar lingkungan kampus berkurang. Selain itu, meskipun penjualan secara *online* sudah dilakukan lewat WhatsApp, Instagram, dan Shopee, platform-platform tersebut belum digunakan dengan baik sebagai cara utama untuk menjual barang. Pelanggan masih lebih suka bertanya terlebih dahulu melalui media *online*, lalu membeli langsung di toko, sehingga kemajuan penjualan secara digital belum mencapai tingkat yang maksimal. Berdasarkan masalah tersebut, dibutuhkan sebuah platform *e-commerce* mandiri yang berbasis *mobile*, sehingga seluruh proses penjualan dapat dilakukan secara digital. Platform ini juga akan membantu memperluas pasar dan mengurangi ketergantungan Ranitha Stuff terhadap *marketplace* yang dimiliki pihak ketiga.

Penelitian sebelumnya sudah membahas pembuatan aplikasi *e-commerce* berbasis *mobile* untuk usaha kecil. Putra et al. [11] mengembangkan aplikasi E-Market berbasis Android untuk mendukung pemasaran produk UMKM di Kecamatan Malingping. Namun, penelitian tersebut belum mengintegrasikan fitur *payment gateway*, chat, pelacakan pesanan, serta rating dan ulasan produk. Putri et al. [1] merancang aplikasi *e-commerce* berbasis Android untuk Toko Suryamart menggunakan *framework* Flutter, namun belum mengintegrasikan *payment gateway* dan fitur *chat* pengguna. Mutia et al. [12] membuat aplikasi *e-commerce* untuk memesan buket bunga berbasis Flutter di Toko Bunga Kisah Kita. Aplikasi ini dilengkapi dengan fitur katalog produk, pembayaran *online*, pelacakan pesanan, dan notifikasi. Namun, mereka menggunakan *backend* berupa PHP tanpa memakai *framework* Laravel, dan belum memasukkan fitur *chat* antar pengguna serta opsi *rating* dan ulasan produk. Jamilah dan Ibrahim [13] mengembangkan aplikasi *e-commerce* furnitur berbasis Android menggunakan Flutter dan Laravel dengan integrasi Midtrans. Namun, penelitian ini masih kurang dalam beberapa fitur, seperti *chat* antara pengguna dan penjual, *rating* dan ulasan produk, serta sistem pelacakan pesanan yang terstruktur. Hal ini secara eksplisit disebutkan sebagai saran pengembangan berikutnya. Karomah dan Arifin [14] membuat aplikasi untuk jual beli sayur menggunakan Flutter, dengan integrasi pembayaran via Midtrans. Namun, aplikasi ini hanya bisa digunakan untuk satu jenis produk pertanian dan belum menyertakan fitur *chatting* atau sistem *rating* dan ulasan. Berdasarkan ketiga penelitian tersebut, terdapat celah penelitian yaitu belum ada sistem *e-commerce* berbasis Flutter yang menggabungkan fitur *payment gateway*, pelacakan pesanan, *chat* pengguna, serta *rating* dan ulasan produk secara lengkap dalam satu platform yang menggunakan Laravel. Sistem ini dirancang untuk memenuhi kebutuhan UMKM dengan berbagai kategori produk.

2. Metode Penelitian

Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *Prototype*. Metode prototipe adalah cara untuk mengembangkan sistem informasi dengan membuat sebuah contoh (prototipe) yang menunjukkan bagaimana sistem tersebut akan terlihat saat dikembangkan lebih lanjut [15]. Metode Prototype digunakan untuk mengembangkan aplikasi *e-commerce* dengan cepat dan teratur [16]. Dengan menggunakan metode prototipe, para pengembang dapat menguji dan memastikan fitur serta fungsi sistem sejak awal. Hal ini membantu dalam memahami kebutuhan pengguna dengan lebih baik dan mengurangi kemungkinan kesalahan dalam ide pengembangan [17].



Gambar 1. Tahapan Metode Prototype

2.1 Metode Pengembangan Sistem

Metode ini merupakan metode pengembangan perangkat lunak yang dilakukan secara iterative dan menghasilkan prototipe sistem yang sesuai dengan kebutuhan *user*. Jika terjadi kesalahan atau ketidaksesuaian dengan *user* pada tahap evaluasi sistem, maka akan kembali ke tahap sebelumnya untuk menyelesaikan masalah atau ketidaksesuaian tersebut [16].

Penelitian ini dilaksanakan selama kurang lebih tiga bulan, yaitu mulai Maret hingga Juni 2026. Proses pengembangan sistem dilakukan secara bertahap sesuai dengan tahapan metode *Prototype*, mulai dari pengumpulan kebutuhan hingga implementasi sistem. Selama proses evaluasi *prototype*, penelitian melibatkan 7 responden yang terdiri atas 1 pemilik UMKM, 1

orang admin, dan 5 orang calon pengguna untuk memperoleh masukan terkait tampilan antarmuka, kemudahan penggunaan, serta fungsionalitas sistem. Masukan yang diperoleh digunakan sebagai dasar dalam penyempurnaan *prototype* sebelum sistem diimplementasikan. Prototyping memiliki beberapa tahapan antara lain:

1. Tahap Pengumpulan Kebutuhan
Tahap ini adalah langkah pertama dalam mengembangkan sistem, di mana dilakukan pengumpulan data dan informasi melalui observasi dan wawancara langsung di toko cabang Ranitha Stuff Blang Pulo. Hasil dari langkah ini adalah daftar kebutuhan sistem, alur proses bisnis, dan fitur-fitur yang akan dibuat dalam aplikasi. Tahap ini membutuhkan durasi waktu selama 2 minggu.
2. Tahap Membangun Prototipe
Berdasarkan kebutuhan yang sudah dikumpulkan sebelumnya, dibangun *wireframe* dan *mockup* tampilan aplikasi menggunakan aplikasi Figma. Prototipe ini berfungsi sebagai contoh awal dari tampilan dan cara kerja aplikasi sebelum masuk ke tahap pengkodean. Tahap ini membutuhkan durasi waktu selama 2 minggu.
3. Tahap Evaluasi Prototipe
Prototipe yang sudah dibuat kemudian dievaluasi bersama dengan perwakilan dari toko cabang Ranitha Stuff di Blang Pulo. Tujuan dari evaluasi ini adalah untuk memastikan bahwa prototipe yang dibuat telah sesuai dengan kebutuhan dan harapan pengguna. Apabila terdapat ketidaksesuaian, akan dilakukan perbaikan dan proses akan kembali ke tahap sebelumnya sampai prototipe dinyatakan sudah sesuai. Tahap ini membutuhkan durasi waktu selama 1 minggu.
4. Tahap Pengkodean Sistem
Prototipe yang telah disetujui selanjutnya diimplementasikan ke dalam bentuk kode program. Pengembangan *frontend mobile* menggunakan *framework* Flutter dengan bahasa pemrograman Dart, *backend* dibangun menggunakan *framework* Laravel dengan bahasa PHP, basis data menggunakan MySQL, serta sistem pembayaran diintegrasikan dengan *payment gateway* Midtrans. Tahap ini membutuhkan durasi waktu selama 4 minggu.
5. Tahap Pengujian Sistem
Sistem yang telah selesai dikodekan selanjutnya diuji menggunakan metode *Black Box Testing* dan *Performance Testing*. Pengujian ini dilakukan untuk memastikan bahwa seluruh fitur dan fungsi dalam aplikasi berjalan sesuai dengan yang diharapkan dan tidak terdapat kesalahan (*error*) dalam alur sistem. Tahap ini membutuhkan durasi waktu selama 2 minggu.
6. Tahap Evaluasi Sistem
Hasil pengujian dievaluasi kembali bersama perwakilan dari toko cabang Ranitha Stuff untuk memastikan bahwa aplikasi yang dibuat sudah memenuhi semua kebutuhan yang telah ditentukan. Jika aplikasi sudah memenuhi semua kebutuhan, maka sistem dinyatakan siap untuk digunakan. Tetapi jika masih ada yang tidak cocok, maka proses akan kembali ke tahap pengkodean untuk melakukan perbaikan. Tahap ini membutuhkan durasi waktu selama 1 minggu.
7. Tahap Penggunaan Sistem
Sistem yang telah berhasil melewati seluruh tahap pengujian dan evaluasi selanjutnya diserahkan sebagai prototipe aplikasi *e-commerce* yang siap digunakan oleh UMKM Ranitha Stuff. Tahap ini membutuhkan durasi waktu selama 1 minggu.

2.2 Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian ini terdiri dari tiga metode, yaitu observasi, wawancara, dan studi pustaka. Ketiga teknik ini dilakukan untuk memperoleh data dan informasi yang dibutuhkan dalam proses pengembangan aplikasi *e-commerce* pada UMKM Ranitha Stuff.

1. Observasi

Pengamatan dilakukan dengan cara mengunjungi langsung toko cabang Ranitha Stuff yang berada di Blang Pulo. Dengan mengunjungi langsung, peneliti bisa melihat keadaan toko dan proses interaksi yang terjadi antara penjual dan pembeli secara langsung.

2. Wawancara

Wawancara dilakukan secara langsung dengan pemilik dan manajer cabang Ranitha Stuff menggunakan metode semi-terstruktur dengan instrumen berupa daftar lima pertanyaan yang telah disusun sebelumnya. Pertanyaan mencakup target pelanggan, kendala operasional, proses pemesanan, pencatatan transaksi, serta kebutuhan pengembangan sistem. Data hasil wawancara kemudian dianalisis secara deskriptif untuk mengidentifikasi kebutuhan pengguna, permasalahan yang dihadapi, dan solusi yang akan diterapkan pada sistem yang dikembangkan.

3. Studi Pustaka

Studi pustaka dilakukan dengan mempelajari berbagai referensi ilmiah seperti jurnal dan artikel yang didapat melalui sumber digital. Selain itu, juga menggunakan dokumentasi internal yang diperoleh dari hasil observasi dan wawancara yang telah dilakukan sebelumnya.

2.3 Analisis Kebutuhan

Analisis kebutuhan sistem dilakukan berdasarkan hasil observasi dan wawancara yang telah dilaksanakan di toko cabang Ranitha Stuff Blang Pulo. Kebutuhan sistem dibagi menjadi empat kategori sebagai berikut:

1. Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang berkaitan langsung dengan fitur dan fungsi utama yang harus tersedia dalam sistem [18].

Table 1. Kebutuhan Fungsional

No	Fitur	Keterangan
1	Registrasi & Login	Pengguna dapat mendaftarkan akun dan masuk ke aplikasi
2	Manajemen Profil	Pengguna dapat mengelola data profil mereka
3	Kategori Produk	Pengguna dapat melihat produk berdasarkan kategori
4	Daftar Produk	Pengguna dapat melihat seluruh daftar produk yang tersedia
5	Detail Produk	Pengguna dapat melihat informasi lengkap setiap produk
6	Keranjang Belanja	Pengguna dapat menambahkan produk ke keranjang belanja
7	Checkout	Pengguna dapat melakukan proses pemesanan produk
8	Manajemen Pesanan	Admin dapat memproses dan mengelola pesanan masuk
9	Riwayat Pesanan	Pengguna dapat melihat riwayat transaksi sebelumnya
10	Payment gateway	Sistem terintegrasi dengan Midtrans untuk pembayaran digital
11	Pelacakan Pesanan	Pengguna dapat memantau status pengiriman pesanan
12	Chat	Pengguna dapat mengirim pesan kepada admin dan admin dapat membalas
13	Rating & Ulasan	Pengguna dapat memberikan <i>rating</i> dan ulasan pada produk

2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan yang tidak berkaitan langsung dengan fungsi utama sistem, namun berperan penting dalam mendukung kualitas sistem secara keseluruhan [18].

Table 2. Kebutuhan Non-Fungsional

No	Kebutuhan	Keterangan
1	Kompatibilitas	Sistem dapat diakses menggunakan perangkat <i>mobile</i> berbasis Android
2	<i>Usability</i>	Antarmuka dirancang sederhana sehingga memudahkan akses ke seluruh fitur utama.
3	Keamanan	Sistem menerapkan autentikasi pengguna untuk menjaga keamanan data. Proses <i>login</i> dengan autentikasi memiliki rata-rata waktu respons sebesar 2,13 detik.
4	Performa	Waktu respons kurang dari 3 detik dan <i>frame rate</i> 59 FPS

3. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras merupakan spesifikasi minimum perangkat yang digunakan dalam proses pengembangan aplikasi.

Table 3. Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Processor	Intel Core i5 / AMD Ryzen 5 atau lebih tinggi
2	RAM	Minimal 8 GB
3	Storage	Minimal 256 GB SSD
4	Smartphone/Emulator	Perangkat Android minimal versi 8.0 (Oreo)

4. Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak merupakan daftar aplikasi dan *framework* yang digunakan dalam proses pengembangan sistem.

Table 4. Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Keterangan
1	Flutter	<i>Framework</i> utama pengembangan aplikasi <i>mobile</i> berbasis Android
2	Laravel	<i>Framework</i> pengembangan <i>backend</i> sistem
3	MySQL	Sistem manajemen basis data
4	PhpMyAdmin	Pengelolaan basis data secara visual
5	Laragon	Web server lokal selama proses pengembangan
6	Visual Studio Code	Code editor untuk pengelolaan kode program
7	Figma	Perancangan wireframe dan mockup antarmuka aplikasi
8	Postman	Pengujian REST API
9	GitHub	Version control dan kolaborasi tim

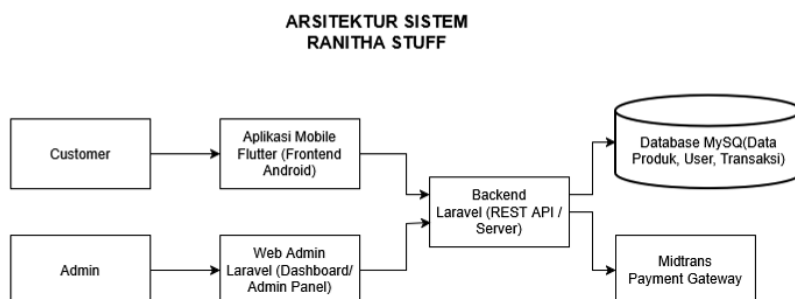
2.4 Perancangan Sistem

Perancangan sistem adalah proses investigasi sistem yang dimulai dengan menemukan masalah dan menggunakan informasi yang diperoleh untuk membuat desain sistem. Tujuan dari perancangan sistem adalah untuk memenuhi kebutuhan pengguna yang ditemukan selama tahap analisis sistem, memberikan gambaran terkait sistem yang akan dibangun, dan memberikan rancang bangun yang jelas bagi orang-orang yang terlibat dalam proses pengembangan sistem [19]. Perancangan sistem ini tidak hanya berfokus pada visualisasi struktur aplikasi, tetapi juga pada hubungan antar komponen dan alur komunikasi data agar sistem dapat berjalan sesuai kebutuhan pengguna. Oleh karena itu, perancangan sistem dijabarkan melalui arsitektur sistem, *use case diagram*, *activity diagram*, dan *class diagram*.

2.4.1 Arsitektur Sistem

Arsitektur sistem adalah gambaran umum mengenai komponen-komponen yang membentuk aplikasi serta bagaimana komponen tersebut saling berinteraksi. Pada penelitian ini,

arsitektur sistem *e-commerce* Ranitha Stuff terdiri dari tiga komponen utama, yaitu aplikasi *mobile* sebagai *frontend*, server/*backend* sebagai pengolah data, dan database sebagai penyimpanan data. Arsitektur sistem yang diterapkan dalam penelitian ini adalah arsitektur *client-server* berbasis *RESTful API* dengan pola MVC (*Model-View-Controller*) pada sisi *backend*. Dengan arsitektur *client-server*, dapat membantu komunikasi yang efektif, sehingga klien dan server bisa berhubungan melalui jaringan seperti Internet atau Intranet [20]. Dalam pendekatan arsitektur MVC, sebuah sistem terbagi menjadi tiga komponen. Model berfungsi sebagai bagian yang mengatur logika, *view* bertugas untuk memberikan antarmuka bagi pengguna, dan *controller* mengelola perubahan yang terjadi pada tampilan [21]. Arsitektur MVC ini merupakan satu kesatuan dalam *framework* Laravel [22]. Dengan adanya arsitektur MVC di Laravel, pengembangan, perawatan, dan pengujian aplikasi web menjadi lebih mudah. Hal ini memungkinkan para pengembang untuk fokus pada setiap komponen tanpa mengganggu komponen lainnya [23]. Pada sisi *frontend*, aplikasi dikembangkan menggunakan *framework* Flutter dengan pola arsitektur MVVM (*Model-View-ViewModel*) yang memisahkan logika bisnis dari tampilan antarmuka pengguna sehingga struktur kode lebih terorganisir dan mudah dikembangkan [24].

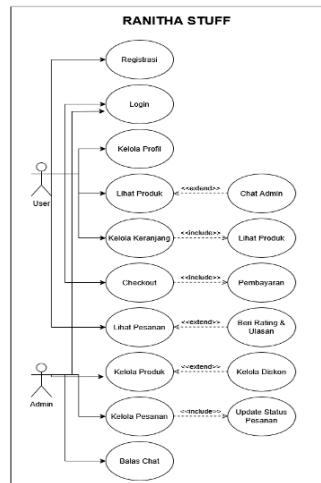


Gambar 2. Arsitektur Sistem Ranitha Stuff

Pada arsitektur ini, alur komunikasi data dimulai dari aplikasi Flutter sebagai *frontend* yang menerima *input* pengguna, kemudian mengirimkan permintaan ke Laravel melalui REST API. Laravel memproses permintaan tersebut, berinteraksi dengan database MySQL untuk membaca atau menyimpan data, lalu mengirimkan hasil dalam bentuk JSON kembali ke aplikasi. Selain itu, integrasi dengan Midtrans digunakan untuk memproses pembayaran dan memperbarui status transaksi secara otomatis pada sistem.

2.4.2 Use Case Diagram

Use Case Diagram adalah alat untuk mengembangkan sistem dengan menjelaskan hubungan antara aktor yang terlibat dalam sistem [25]. *Use Case Diagram* digunakan untuk menjelaskan desain sistem kepada pengguna dan merencanakan semua fitur yang ada dalam sistem yang akan dibuat [19].

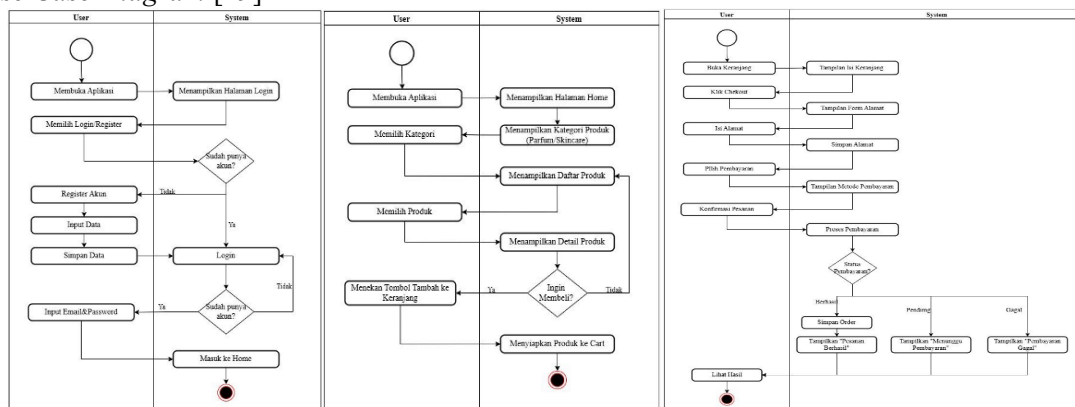


Gambar 3. Use Case Diagram

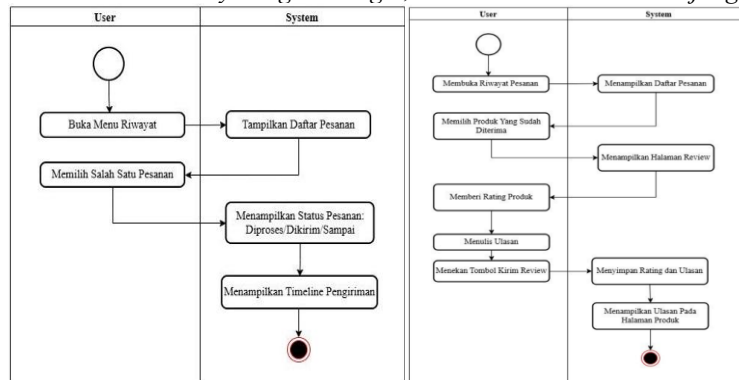
Hubungan antara *Customer* dan Admin pada *use case diagram* menunjukkan pembagian hak akses yang jelas dalam sistem. *Customer* berperan sebagai pengguna utama yang melakukan aktivitas transaksi, sedangkan Admin bertugas mengelola data produk, pesanan, pesan pelanggan, dan notifikasi. Setiap aktivitas yang dilakukan oleh kedua aktor tersebut diproses melalui *backend* sehingga perubahan data dapat langsung tersimpan ke *database* dan ditampilkan sesuai kebutuhan masing-masing pengguna.

2.4.3 Activity Diagram

Activity Diagram menunjukkan proses kerja atau kegiatan yang berlangsung dalam suatu system [25]. *Activity Diagram* dibuat berdasarkan satu atau lebih *Use Case* yang terdapat dalam *Use Case Diagram* [19].



Gambar 4. Activity Diagram Login, Memilih Produk & Keranjang



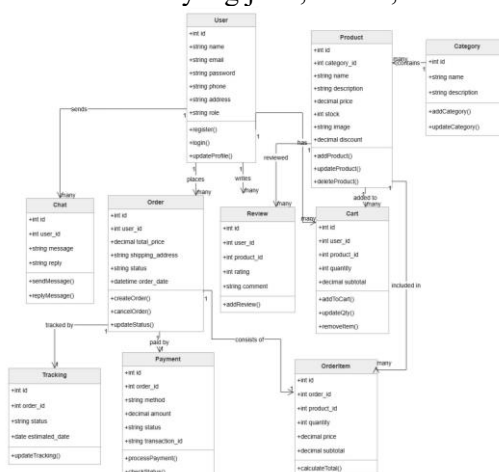
Gambar 5. Activity Diagram Tracking Pesanan dan Rating & Ulasan

Setiap *activity diagram* menggambarkan alur proses yang saling terhubung antara pengguna, *backend*, *database*, dan layanan pembayaran. Pada proses *login*, data autentikasi

divalidasi melalui *backend* sebelum pengguna diarahkan ke halaman utama. Pada proses *checkout*, data keranjang dan alamat diproses oleh *backend*, lalu diteruskan ke Midtrans untuk pembayaran. Pada proses *tracking* pesanan, sistem mengambil status pengiriman dari *database* dan menampilkannya secara *real-time* kepada pengguna. Sedangkan pada proses *rating* dan ulasan, data yang dimasukkan pengguna disimpan melalui *backend* dan ditampilkan kembali pada halaman produk.

2.4.4 Class Diagram

Class Diagram kelas adalah gambar yang menunjukkan struktur dan penjelasan tentang kelas, paket, dan objek serta bagaimana mereka saling berhubungan, seperti pengendalian, pewarisan, asosiasi, dan lainnya [25]. Penggunaan Diagram Kelas dalam merancang sistem diharapkan dapat menciptakan model sistem yang jelas, teratur, dan mudah dimengerti [26].



Gambar 6. Class Diagram

Class diagram menunjukkan hubungan antar entitas yang mendukung proses bisnis aplikasi. Entitas *User* berhubungan dengan *Order*, *Cart*, *Review*, dan *Message* karena satu pengguna dapat melakukan transaksi, memberi ulasan, dan berkomunikasi dengan admin. Sementara itu, *Product* berelasi dengan *Category* dan *Order Detail* untuk mendukung proses pemilihan barang hingga transaksi selesai. Relasi tersebut memastikan bahwa data pengguna, transaksi, dan komunikasi tersusun secara konsisten dalam *database* MySQL dan dikelola melalui *backend* Laravel.

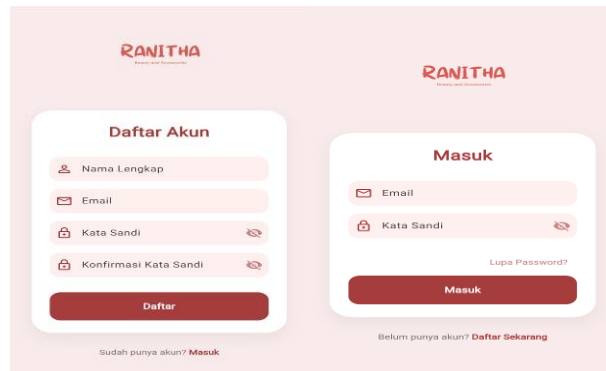
3. Hasil dan Pembahasan

Hasil penelitian ini berupa aplikasi *mobile* Ranitha Stuff yang dikembangkan untuk mendukung penjualan produk UMKM kosmetik dan aksesoris. Berdasarkan hasil pengujian, seluruh fitur pada aplikasi dapat berjalan dengan baik sesuai dengan kebutuhan pengguna.

3.1 Implementasi Sistem User

1. Halaman Login dan Register

Gambar di bawah ini menampilkan halaman registrasi dan *login* pada aplikasi Ranitha Stuff. Pengguna mengakses halaman registrasi untuk membuat akun baru dengan mengisi informasi pribadi mereka, sedangkan halaman *login* digunakan untuk masuk ke aplikasi dengan menggunakan email dan kata sandi yang sudah didaftarkan [27].

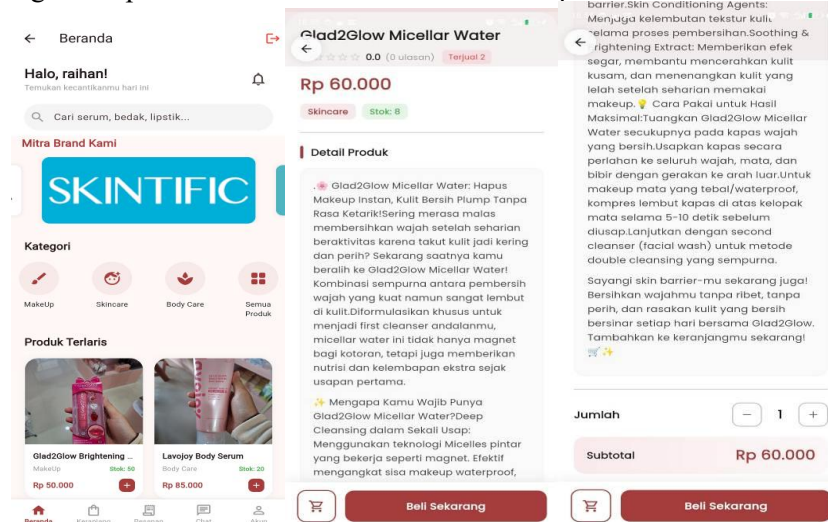


Gambar 7. Tampilan Halaman Login dan Register

Proses autentikasi dilakukan melalui backend Laravel menggunakan REST API, kemudian data pengguna divalidasi sebelum pengguna diarahkan ke halaman utama aplikasi.

2. Halaman Daftar Produk dan Detail Produk

Halaman daftar produk dan detail produk digunakan untuk menampilkan informasi lengkap mengenai produk. Dengan cara ini, pengguna dapat melihat nama produk, harga, deskripsi, dan gambar produk sebelum mereka membuat keputusan untuk membeli [28].

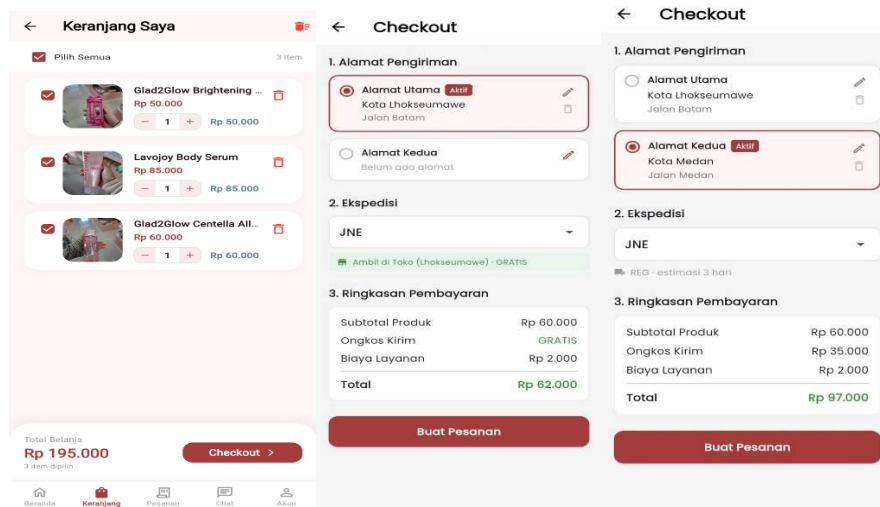


Gambar 8. Tampilan Daftar Produk dan Detail Produk

Data produk diambil dari *database* MySQL melalui REST API sehingga informasi produk, kategori, serta ulasan dapat ditampilkan secara lengkap pada aplikasi.

3. Halaman Keranjang dan Checkout

Gambar di bawah ini menampilkan halaman keranjang dan *Checkout* pada aplikasi Ranitha Stuff. Halaman keranjang menampilkan produk yang dipilih pengguna, sedangkan halaman *Checkout* digunakan untuk melakukan konfirmasi pesanan dan pembayaran.

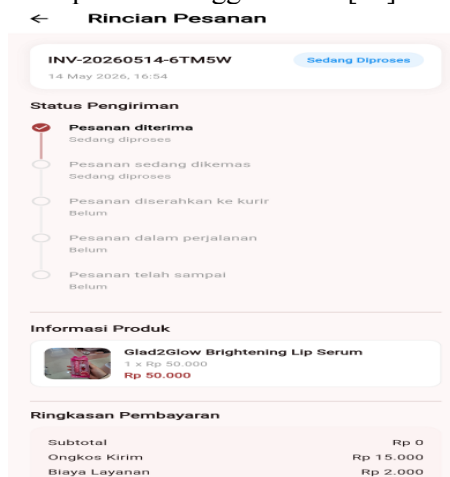


Gambar 9. Tampilan Keranjang dan Checkout

Pada proses *checkout*, *backend* memvalidasi data pesanan dan mengintegrasikan transaksi dengan *payment gateway* Midtrans sebelum status pembayaran diperbarui ke dalam sistem.

4. Halaman *Tracking* Pesanan

Gambar di bawah ini menampilkan halaman untuk melacak pesanan di aplikasi Ranitha Stuff. Halaman ini digunakan untuk menunjukkan informasi mengenai status pesanan, sehingga pengguna dapat memantau proses pesanan hingga selesai [29].

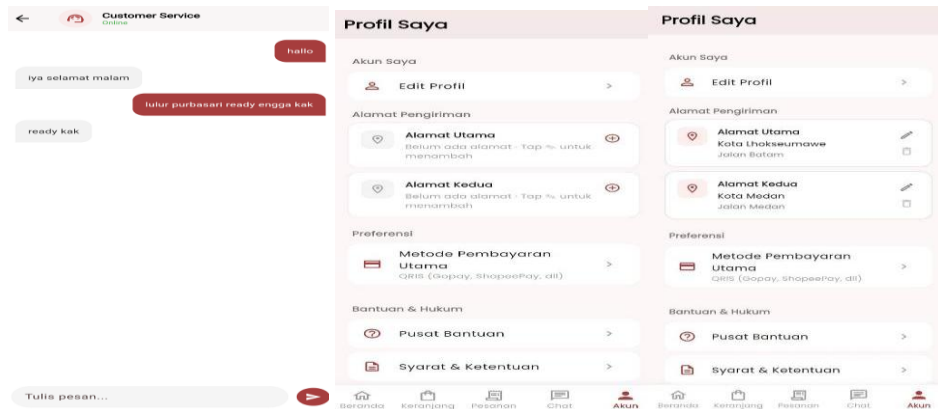


Gambar 10. Tampilan Tracking Pesanan

Informasi status pesanan diperoleh dari *database* melalui REST API sehingga pengguna dapat memantau perkembangan pesanan secara *real-time*.

5. Halaman *Chat* Admin dan Profil Pengguna

Gambar di bawah ini menampilkan halaman *chat* admin dan profil pengguna pada aplikasi Ranitha Stuff. Halaman *chat* admin digunakan untuk berkomunikasi dengan admin, sedangkan halaman profil pengguna digunakan untuk melihat dan mengubah informasi akun.



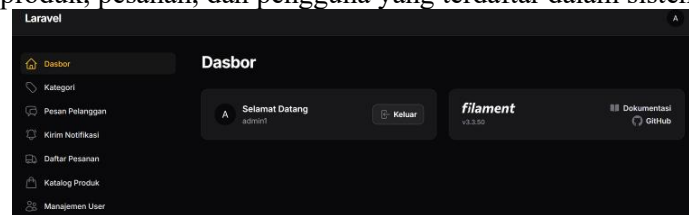
Gambar 11. Tampilan Chat Admin dan Profil Pengguna

Seluruh pesan yang dikirim pengguna diproses melalui *backend* Laravel dan disimpan pada *database* sehingga komunikasi antara pengguna dan admin dapat dilakukan dalam satu aplikasi.

3.2 Implementasi Sistem Admin

1. Halaman *Dashboard* Admin

Gambar di bawah ini menunjukkan dashboard admin yang memuat ringkasan informasi mengenai jumlah produk, pesanan, dan pengguna yang terdaftar dalam sistem [30].

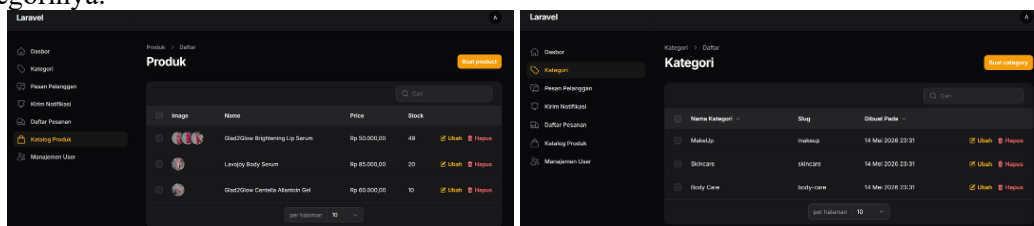


Gambar 12. Tampilan Dashboard Admin

Data pada *dashboard* diperoleh dari *database* secara *real-time* sehingga admin dapat memantau aktivitas sistem dengan lebih mudah.

2. Halaman Katalog Produk dan Kategori

Gambar di bawah ini menampilkan halaman katalog produk dan kategori. Halaman ini digunakan oleh admin untuk menambah, mengubah, dan menghapus data produk beserta kategorinya.

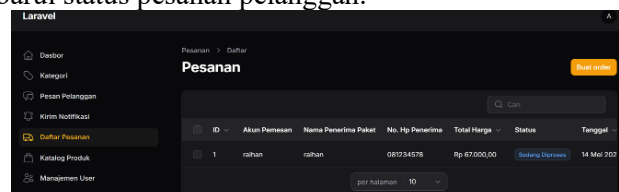


Gambar 13. Tampilan Katalog Produk dan Kategori

Pengelolaan data produk dilakukan menggunakan fungsi CRUD yang diproses oleh backend Laravel sebelum disimpan ke *database*.

3. Halaman Daftar Pesanan

Gambar di bawah ini menampilkan halaman daftar pesanan yang digunakan admin untuk melihat dan memperbarui status pesanan pelanggan.

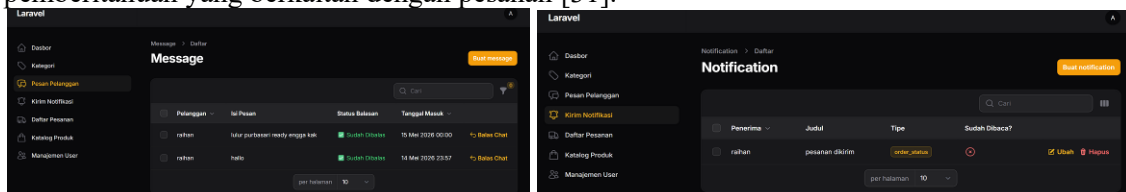


Gambar 14. Tampilan Daftar Pesanan

Perubahan status pesanan akan langsung diperbarui pada sistem sehingga informasi yang diterima pengguna selalu sesuai dengan kondisi terbaru.

4. Halaman Pesan Pelanggan dan Kirim Notifikasi

Gambar di bawah ini menampilkan halaman notifikasi dan halaman pesan pelanggan. Fitur ini digunakan untuk berkomunikasi dengan pengguna dan mengirimkan informasi atau pemberitahuan yang berkaitan dengan pesanan [31].



Gambar 18. Tampilan Pesan Pelanggan dan Kirim Notifikasi

Fitur ini mendukung komunikasi antara admin dan pengguna serta penyampaian informasi terkait proses transaksi.

3.3 Pengujian Sistem

Pengujian perangkat lunak sangat penting untuk memastikan bahwa kualitas perangkat lunak yang dibuat terjamin dan tidak ada kesalahan yang terjadi [32]. Metode *Black Box Testing* digunakan untuk menguji sistem untuk memastikan bahwa setiap fitur aplikasi berfungsi sesuai dengan kebutuhan pengguna [33]. Pengujian dilakukan dengan menguji fungsi-fungsi utama tanpa melihat kode program, melainkan berdasarkan masukan dan keluaran yang dihasilkan oleh sistem.

Table 5. Hasil Black Box Testing

No	Fitur yang Diuji	Hasil yang Diharapkan	Status
1	Registrasi	Akun baru berhasil dibuat dan pengguna dapat masuk ke aplikasi	Berhasil
2	Login	Pengguna berhasil masuk ke dalam sistem	Berhasil
3	Lihat Produk	Daftar produk kosmetik dan aksesoris tampil dengan lengkap	Berhasil
4	Keranjang	Produk berhasil masuk ke keranjang belanja	Berhasil
5	Checkout	Pesanan berhasil dibuat dan data transaksi tersimpan	Berhasil
6	Tracking	Informasi status pesanan dapat dilihat oleh pengguna	Berhasil
7	Chat Admin	Pesan berhasil terkirim dan diterima oleh admin	Berhasil
8	Kelola Produk	Admin dapat menambah, mengubah, dan menghapus produk	Berhasil
9	Kelola Pesanan	Admin dapat memperbarui status pesanan	Berhasil
10	Rating & ulasan	Penilaian dan komentar berhasil disimpan	Berhasil

Selain *Black Box Testing*, penelitian ini juga melakukan *Performance Testing* untuk mengevaluasi kinerja aplikasi pada sisi *backend* dan *frontend*. Pengujian *backend* dilakukan menggunakan Postman dengan mengukur waktu respons beberapa *endpoint* utama, sedangkan pengujian *frontend* dilakukan menggunakan Flutter DevTools untuk mengukur performa *rendering* aplikasi saat dijalankan pada perangkat Android.

Table 6. Performance Testing API

Endpoint	Response Time	Keterangan
/api/login	2,13 detik	Baik
/api/products	1,77 detik	Baik
/api/checkout	4,67 detik	Cukup

Table 7. Performance Testing Frontend

Metrik	Hasil	Target	Keterangan
Frame Rate	59 FPS	≥ 55 FPS	Baik
Jank Frame	2	0-5	Baik

3.4 Pembahasan

Berdasarkan hasil implementasi dan pengujian, aplikasi *mobile* Ranitha Stuff berhasil mendukung proses penjualan produk UMKM di bidang kosmetik dan aksesoris. Seluruh sepuluh fitur yang diuji menggunakan metode *Black Box Testing* berhasil berjalan tanpa ditemukan kesalahan, yang menunjukkan bahwa sistem telah berfungsi sesuai dengan kebutuhan yang ditentukan. Aplikasi ini memungkinkan pengguna untuk melihat produk, melakukan pemesanan, melakukan pembayaran melalui *payment gateway* Midtrans yang terintegrasi, serta memantau status pesanan secara *real-time*. Sementara itu, admin dapat mengelola produk, pesanan, dan komunikasi dengan pelanggan melalui satu platform yang terintegrasi.

Dibandingkan dengan penelitian sebelumnya, aplikasi yang dikembangkan dalam penelitian ini menawarkan fitur yang lebih lengkap. Mutia dan timnya [12] mengembangkan aplikasi *e-commerce* berbasis Flutter, namun belum mengintegrasikan fitur *chat* serta sistem *rating* dan ulasan produk. Jamilah dan Ibrahim [13] mengembangkan aplikasi serupa menggunakan Flutter dan Laravel, tetapi belum menyediakan fitur pelacakan pesanan yang terstruktur dan komunikasi *chat* antara pengguna dan penjual. Karomah dan Arifin [14] membangun aplikasi jual beli berbasis Flutter dengan integrasi Midtrans, namun masih terbatas pada satu kategori produk. Berbeda dengan penelitian-penelitian tersebut, penelitian ini berhasil mengintegrasikan fitur *payment gateway*, pelacakan pesanan, *chat*, serta *rating* dan ulasan produk dalam satu platform berbasis Flutter dan Laravel yang dirancang khusus untuk UMKM dengan berbagai kategori produk. Hasil penelitian ini menunjukkan bahwa aplikasi yang dikembangkan mampu mengatasi keterbatasan yang ditemukan pada penelitian sebelumnya dan memberikan solusi yang lebih lengkap untuk mendukung kebutuhan penjualan digital UMKM.

Selain itu, hasil pengembangan ini menunjukkan bahwa aplikasi *e-commerce* berbasis Android dapat menjadi solusi yang lebih efektif bagi UMKM dalam mengelola penjualan digital secara mandiri. Integrasi fitur seperti *payment gateway*, pelacakan pesanan, *chat*, *rating*, dan ulasan tidak hanya meningkatkan kelengkapan fungsi sistem, tetapi juga mendukung interaksi yang lebih baik antara penjual dan pembeli. Dengan demikian, sistem ini berpotensi membantu meningkatkan efisiensi operasional, mempercepat proses transaksi, serta memperluas jangkauan pasar UMKM Ranitha Stuff, terutama pada periode penurunan penjualan seperti libur semester.

4. Kesimpulan

Penelitian ini telah berhasil merancang dan membangun aplikasi *e-commerce* berbasis Flutter pada UMKM Ranitha Stuff yang bergerak di bidang kosmetik, aksesoris, dan perawatan kulit. Aplikasi dikembangkan menggunakan *framework* Flutter pada sisi *frontend*, Laravel pada sisi *backend*, MySQL sebagai *database*, serta integrasi *payment gateway* Midtrans untuk mendukung transaksi digital. Fitur-fitur yang berhasil diimplementasikan meliputi registrasi dan *login* pengguna, manajemen produk, keranjang belanja, *Checkout*, pelacakan pesanan, *chat* pengguna dengan admin, serta *rating* dan ulasan produk.

Berdasarkan hasil pengujian menggunakan metode *Black Box Testing* terhadap 10 fitur utama aplikasi, seluruh fitur berhasil berjalan sesuai dengan kebutuhan pengguna dengan tingkat keberhasilan mencapai 100%. Hasil *Performance Testing* menunjukkan rata-rata *response time* API sebesar 2,86 detik dan *frame rate* aplikasi mencapai 59 FPS. Hasil tersebut menunjukkan bahwa aplikasi mampu menjalankan proses transaksi dan pengelolaan data secara stabil sehingga berpotensi meningkatkan efisiensi proses penjualan digital pada UMKM Ranitha Stuff. Dengan demikian, aplikasi yang dikembangkan mampu membantu UMKM mengurangi ketergantungan terhadap *marketplace* pihak ketiga melalui platform penjualan terintegrasi.

Untuk pengembangan selanjutnya, aplikasi dapat ditingkatkan dengan menambahkan fitur rekomendasi produk, diskon dan *flash sale*, integrasi dengan jasa pengiriman ekspedisi secara otomatis, serta pengembangan versi iOS agar jangkauan pengguna semakin luas.

Daftar Pustaka

- [1] H. Adilla Syafira Putri, Ade Eviyanti, “Rancang Bangun Aplikasi E-Commerce Berbasis Android Pada Toko Suryamart Menggunakan Framework Flutter,” *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 3, pp. 257–265, 2023.
- [2] E. Sulaksana, A. Ichwani, N. Anwar, and P. Setiawati, “Aplikasi E-Commerce Pakaian Berbasis Android Menggunakan Metode Prototype,” *METHOMIKA J. Manaj. Inform. Komputerisasi Akunt.*, vol. 7, no. 2, pp. 307–318, 2023.
- [3] M. Sari and S. Andriasari, “Perancangan Aplikasi Mobile E-Commerce Berbasis Android pada Natasya Butik,” *Remik*, vol. 7, no. 1, pp. 444–451, 2023, doi: 10.33395/remik.v7i1.12060.
- [4] R. S. Khairunnisa and H. Siregar, “Analisis Sistematis Dampak E-Commerce Mobile Platform terhadap Profitabilitas UMKM Digital,” *Infomatek*, vol. 28, no. 1, pp. 9–22, 2026, doi: 10.23969/infomatek.v28i1.41940.
- [5] I. A. dan Dimas Gilang Pratama and R. Prabowo, “Inovasi Modul Pembelian untuk E-Commerce UMKM: Solusi Berbasis Android,” *J. Pepadun*, vol. 4, no. 1, pp. 308–314, 2023, doi: 10.23960/pepadun.v4i1.139.
- [6] D. M. Risda Choirunisa, “Literature Review: Peran E-Commerce Dalam Meningkatkan Pendapatan Umkm Di Era New Normal,” *J. Ekon. Bisnis dan Ind.*, vol. 5, no. 1, pp. 9–15, 2023, doi: 10.52061/ebi.v7i2.461.
- [7] BADAN PUSAT STATISTIK, “Statistik E-Commerce 2024,” JAKARTA, 2025.
- [8] D. M. Heri Sudibyo, “Implementasi E-Commerce Menggunakan Framework Technology Organization Enviroment Untuk Peningkatan Pemasaran Dan Penjualan Produk Pada Usaha Kecil Menengah (Ukm) Dharmasraya,” *JOISIE J. Inf. Syst. Informatics Eng.*, vol. 6, no. 2, pp. 64–68, 2022.
- [9] R. H. P. S. Nurrohmi Zaki, “Implementasi Aplikasi Android dalam Sistem Restock UMKM Maju Jaya Accessories,” *J. Indones. Manaj. Inform. dan Komun.*, vol. 6, no. 1, pp. 318–333, 2025, doi: 10.35870/jimik.v6i1.1211.
- [10] N. N. K. S. Candra Wijaya, Felik Rolantius Jovito, Viktor Handrianus Pranatawijaya, “Pemanfaatan Dart, Flutter, Firebase, Dan Stripe Dalam Membangun Aplikasi Mobile,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 8, no. 3, pp. 4017–4025, 2024.
- [11] A. S. Putra, A. Sukendar, and W. Gusdya, “Pembangunan Aplikasi E-Market UMKM Berbasis Android Menggunakan Framework Flutter,” *Pasinformatik*, vol. 4, no. 1, 2025.
- [12] R. Acep Sandi Mutia, Sarji, Asep Munawar, Teguh Aljauza, Tedi Budiman, “Perancangan dan Pengembangan Aplikasi E-Commerce untuk Pemesanan Buket Bunga Berbasis Flutter : Studi Kasus Toko Bunga Kisah Kita,” *Bul. Sist. Inf. dan Teknol. Islam*, vol. 6, no. 2, pp. 58–66, 2025.
- [13] A. A. I. Jamilah, “Rancang bangun aplikasi e-commerce furnitur berbasis android,” *J. JUIT*, vol. 2, no. 2, pp. 62–72, 2023.
- [14] S. A. M. Muzaki Syahrul Karomah, “Aplikasi Jual Beli Sayur Berbasis Android Integrasi,” *STIMATA*, vol. 8, pp. 519–531, 2026.
- [15] S. Abdullah Azam, “Pemodelan UML dan Perancangan Prototype Aplikasi Berbasis Web di Toko Herbal Albiruni,” *J. Manaj. Sist. Inf. dan Teknol.*, vol. 14, no. 2, pp. 73–79, 2024.
- [16] Rahmandhika, “Rancang Bangun Website E-commerce Menggunakan Metode Prototyping dan Algoritma Insertion Sorting (Studi Kasus : PT Bumi Baureksa Pratama),” 2023.
- [17] P. Kustanto, B. K. Ramadhan, and A. Noe, “Penerapan Metode Prototype dalam Perancangan Media Pembelajaran Interaktif,” *J. JSRCS*, vol. 5, no. 1, pp. 83–94, 2024.

- [18] dan T. O. Novenda Kartika Putrianto, Yuanita Maria Widyastuti, “Perencanaan dan analisis kebutuhan pengguna dalam pengembangan sistem informasi hubungan pelanggan,” *KURAWAL J. Teknol. Inf. dan Ind.*, vol. 7, no. 1, pp. 1–11, 2024.
- [19] S. Narulita, A. Nugroho, and M. Z. Abdillah, “Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS) Universitas Nasional Karangturi Semarang , Indonesia (deskripsi) dan perancangan sistem , khususnya pada pemrogr,” *Bridg. J. Publ. Sist. Inf. dan Telekomun.*, vol. 2, no. 3, pp. 244–256, 2024.
- [20] G. Nyabuto, “Client-server Architecture , a Review,” *Int. J. Adv. Sci. Comput. Appl.*, vol. 3, no. 2, pp. 1–4, 2024, doi: 10.47679/ijasca.v3i2.48.
- [21] P. Tamir, I. Permana, and A. N. Sihanato, “Implementasi Arsitektur MVC Dalam Pengembangan Aplikasi Customer Relationship Portal,” *J. Teknol. Inf.*, vol. 10, no. 1, pp. 50–57, 2024.
- [22] F. A. Galih Surono, Yogasetya Suhandi, “Penerapan MVC Arsitektur Pada Sistem Informasi Monitoring Pada Divisi Produksi Menggunakan Laravel Framework,” *J. Sensi*, vol. 8, no. 2, pp. 180–189, 2022.
- [23] S. Levianita Rahmawati, “Desain Pengembangan Website dengan Arsitektur Model View Controller pada Framework Laravel,” *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 4, pp. 785–790, 2024.
- [24] T. N. Julianto Royand Wahyudi Simanullang, Fransiskus Zoromi, Unang Rio, “Aplikasi penjualan produk it dan jasa service berbasis android dengan metode mvvm,” *J. TEKINKOM*, vol. 7, no. 1, pp. 62–71, 2024, doi: 10.37600/tekinkom.v7i1.1241.
- [25] F. R. M. Rhifky Wayahdi, “Pemodelan Sistem Penerimaan Anggota Baru dengan Unified Modeling Language (UML) (Studi Kasus : Programmer Association of Battuta),” *J. Minfo Polgan*, vol. 12, no. 1, pp. 1514–1521, 2023.
- [26] S. W. Ramdany, S. A. Kaidar, B. Aguchino, C. Amelia, and A. Putri, “Penerapan UML Class Diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web,” *J. Ind. Eng. Syst. e-ISSN*, vol. 5, no. 1, pp. 30–41, 2024.
- [27] M. Saad, N. Ishlah, N. S. R. I. Wahyuni, and L. Karlitasari, “Perancangan Antarmuka Pengguna Aplikasi Mobile Sistem Informasi Akademik (SIMAK) menggunakan Metode Design Thinking (Studi Kasus : SIMAK Universitas Pakuan) User Interface Design for the Academic Information System,” *JIKA-Jurnal Ilmu Komput. Agri-informatika*, vol. 11, no. 1, pp. 86–99, 2024.
- [28] R. Manurung, T. T. Sipahutar, and B. R. M. Nainggolan, “Sistem Informasi Penjualan Terintegrasi Android: Solusi Digitalisasi Umkm Dalam Era Ekonomi Digital (Studi Kasus: Kugar Minamas Pansela),” *Electro Luceat*, vol. 10, no. 2, pp. 1–9, 2024, doi: 10.32531/jelekn.v10i2.843.
- [29] F. Faturohman, S. Widodo, A. Ramadhany, and S. D. Nurfitriadi, “Implementasi White Label Dalam Reuse Aplikasi E-Commerce Berbasis Mobile Untuk Umkm,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 11, no. 2, pp. 750–757, 2026, doi: 10.29100/jupi.v11i2.8025.
- [30] M. Irfan, M. Ushbah, and H. Sentyaki, “Development of an AI-Assisted Mobile Point of Sales and E- Commerce System with Real-Time Inventory Synchronization,” *Media J. Inform.*, vol. 18, no. 1, pp. 49–60, 2026.
- [31] N. P. Kevin Antony, “Analisis Peran Push Notification dalam Mendorong Pembelian Peralatan Rokok Elektrik di E-Commerce,” *J. Untar*, vol. 3, no. 2, pp. 230–239, 2023.
- [32] Mintarsih, “Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation,” *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 1, pp. 33–35, 2023.
- [33] E. Novalia and A. Voutama, “Black Box Testing dengan Teknik Equivalence Partitions Pada Aplikasi Android M-Magazine Mading Sekolah,” *Syntax J. Inform.*, vol. 11, no. 01, pp. 23–35, 2022, doi: 10.35706/syji.v11i01.6413.