

# IMPLEMENTASI FITUR DESKRIPSI PRODUK BERBASIS API GEMINI DALAM PENGEMBANGAN E-COMMERCE BERBASIS MOBILE MENGGUNAKAN FRAMEWORK FLUTTER

Cindi Lusya Kiareni, Cinda Sorisa, Nova Noor Kamala Sari, Viktor Handrianus Pranatawijaya

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya

Jalan Hendrik Timang, Palangka Raya, Kalimantan Tengah, Indonesia

cindilusiakiareni@gmail.com

## ABSTRAK

Dengan kemajuan teknologi informasi, e-commerce telah meningkat secara signifikan karena kebutuhan akan kemudahan akses melalui perangkat mobile. Solusi untuk mengubah e-commerce menjadi format berbasis mobile yang responsif diperlukan karena platform berbasis web atau desktop tidak lagi memadai. Untuk meningkatkan efisiensi pengelolaan deskripsi produk, penelitian ini mengembangkan aplikasi e-commerce berbasis mobile yang menggunakan framework *Flutter* dan menggunakan *Google Generative AI* menggunakan *API Gemini*. Metode penelitian ini mencakup desain aplikasi dan perancangan sistem dengan diagram alur kerja untuk memastikan integrasi *API Gemini*. Hasilnya menunjukkan bahwa mengubah tampilan website e-commerce menjadi versi mobile meningkatkan efisiensi pengelolaan deskripsi produk. Hasil uji fungsionalitas, responsivitas, dan kinerja aplikasi memuaskan. Penelitian ini diharapkan dapat membantu pengembangan aplikasi e-commerce yang lebih responsif dan efisien.

**Kata kunci:** *API Gemini, Google Generative AI, E-commerce, Flutter, Aplikasi Mobile*

## 1. PENDAHULUAN

Perkembangan teknologi informasi merujuk pada kemajuan dan inovasi dalam ranah teknologi yang terkait dengan pengumpulan, pengolahan, penyimpanan, dan penyebaran informasi. Dampak positif dari kemajuan ini termanifestasi dalam kemudahan yang diberikan serta peningkatan efisiensi dalam berbagai kegiatan organisasi. Contohnya, penerapan teknologi informasi sebagai sarana komunikasi elektronik yang memungkinkan peningkatan layanan kepada masyarakat [1].

Di era digital saat ini, e-commerce telah menjadi bagian integral dalam kehidupan masyarakat. Pertumbuhannya yang pesat dalam beberapa dekade terakhir mendorong platform e-commerce menjadi salah satu cara utama untuk berbelanja, baik bagi konsumen maupun bisnis. Namun, dengan semakin maraknya penggunaan perangkat mobile, website e-commerce yang dirancang untuk desktop saja tidak lagi cukup untuk memenuhi kebutuhan pengguna. Diperlukan solusi yang memungkinkan pengguna untuk mengakses e-commerce dengan mudah dan nyaman di perangkat mobile mereka.

Tujuan penelitian ini adalah mengembangkan aplikasi e-commerce berbasis mobile menggunakan framework *Flutter* serta mengintegrasikan fitur deskripsi produk berbasis *Google Generative AI* menggunakan *API Gemini* untuk meningkatkan efisiensi pengelolaan produk. Penelitian ini bertujuan untuk memastikan responsivitas aplikasi di berbagai perangkat, meningkatkan aksesibilitas pengguna melalui antarmuka yang responsif, serta menyediakan solusi yang memungkinkan pengelolaan deskripsi produk yang konsisten dan up-to-date secara otomatis.

Penelitian ini membahas implementasi AI dalam pengembangan aplikasi e-commerce, dengan fokus pada *API Gemini*. *API Gemini* memungkinkan otomatisasi fungsi penting seperti pembuatan deskripsi produk berdasarkan data yang ada, meningkatkan efisiensi dan konsistensi informasi. *API* ini mendukung integrasi mudah dengan platform e-commerce, memastikan deskripsi produk selalu up-to-date dan menarik.

Penelitian ini diharapkan berkontribusi pada pengembangan aplikasi e-commerce berbasis mobile, serta menunjukkan bagaimana *API Gemini* meningkatkan efisiensi pengelolaan deskripsi produk.

## 2. TINJAUAN PUSTAKA

### 2.1. E-commerce

E-commerce telah menjadi bagian integral dari kehidupan sehari-hari, dengan pertumbuhan yang signifikan dalam beberapa dekade terakhir. Dengan meningkatnya penggunaan internet dan perangkat mobile, platform e-commerce telah berkembang menjadi salah satu cara utama untuk berbelanja, baik untuk konsumen maupun bisnis. E-commerce, sering dikenal sebagai perdagangan elektronik, mengacu pada pembelian dan penjualan barang dan jasa menggunakan internet atau platform digital lainnya. Perdagangan online merupakan teknologi bisnis baru di mana produk diperdagangkan secara online atau melalui sarana elektronik oleh penjual [2]. Perdagangan elektronik menjadi semakin lazim, karena ada kebutuhan bagi penjual dan pembeli untuk menyelesaikan transaksi [3].

## 2.2. Aplikasi Mobile

Aplikasi mobile merupakan perangkat lunak yang dirancang khusus untuk dijalankan pada perangkat mobile seperti smartphone atau tablet. Aplikasi ini dapat mengakses berbagai fitur dan layanan perangkat mobile, seperti kamera, lokasi, sensor gerak, dan lainnya, serta dapat terhubung dengan internet untuk mengakses konten dan layanan online. Singkatnya, aplikasi adalah sebuah program yang berjalan pada sebuah perangkat untuk menjalankan fungsi tertentu untuk pengguna atau aplikasi lain, dan dapat digunakan dengan skrip tertentu. Di sisi lain, perangkat seluler dapat dianggap sebagai alat transportasi dari satu lokasi ke lokasi lain [4]. Aplikasi mobile, juga dikenal sebagai aplikasi untuk perangkat bergerak, adalah aplikasi dalam bentuk aplikasi yang dikembangkan dengan menggunakan program komputer untuk digunakan pada perangkat bergerak seperti tablet, smartphone, dan jam tangan pintar [5].

## 2.3. Android Studio

Android Studio adalah sebuah lingkungan pengembangan terpadu (IDE) sumber terbuka untuk membuat aplikasi Android. Dibuat oleh Google dan dibangun di atas IntelliJ IDEA, Android Studio menawarkan berbagai fitur dan alat yang dibutuhkan pengembang untuk membuat aplikasi Android yang inovatif dan berkualitas tinggi. Android adalah sistem operasi seluler yang dikembangkan oleh Google yang dirancang khusus untuk perangkat seluler seperti tablet dan smartphone. Sebagai salah satu sistem operasi mobile paling populer di dunia, Android menawarkan lingkungan yang kuat dan fleksibel bagi pengembang untuk menciptakan berbagai jenis aplikasi dan pengalaman pengguna. Android SDK menyediakan alat dan kerangka kerja untuk pemrograman Java (API) yang dibutuhkan untuk mengembangkan aplikasi menggunakan bahasa pemrograman Java pada platform Android [2]. Seiring perkembangan zaman, android menjadi sebuah device yang sigap dalam berinovasi [6]. Android merupakan subset perangkat lunak untuk perangkat mobile yang meliputi sistem operasi, *middleware*, dan aplikasi inti yang dirilis oleh Google [1].

## 2.4. Flutter

Flutter adalah sebuah framework open-source yang dikembangkan oleh Google untuk membangun antarmuka pengguna (UI) aplikasi mobile. Flutter menggunakan bahasa pemrograman Dart dan menyediakan berbagai widget UI yang kaya dan dapat disesuaikan untuk membuat aplikasi mobile yang indah dan responsif. Proyek Flutter yang ditulis dalam bahasa pemrograman Dart dan memiliki fitur AOT (*Ahead-of-Time*) memungkinkannya untuk digunakan tanpa perlu diterjemahkan lebih lanjut ke platform arsitektur asli [4].

## 2.5. Dart

Dart adalah bahasa pemrograman yang digunakan oleh Flutter untuk mengembangkan aplikasi mobile, web, dan server. Dirancang oleh Google, Dart merupakan bahasa yang modern, efisien, dan mudah dipelajari. Dalam tinjauan pustaka ini, kita akan membahas pengertian, fitur, dan kelebihan Dart sebagai landasan teori untuk pengembangan aplikasi dengan Flutter. Dart adalah bahasa tingkat tinggi yang digunakan dalam manajemen memori otomatis (*garbage collection*), yang membuat aplikasi dapat berjalan lebih cepat di banyak platform [7]. Dart merupakan bahasa untuk mengembangkan aplikasi modern yang mudah digunakan, memiliki kinerja implementasi yang tinggi, dan dapat digunakan tanpa perlu melakukan kompilasi secara konstan [8].

## 2.6. Kecerdasan Buatan

Kecerdasan Buatan (*Artificial Intelligence, AI*) adalah cabang dari ilmu komputer yang berfokus pada pembuatan sistem yang mampu melakukan tugas-tugas yang biasanya membutuhkan kecerdasan manusia. Tugas-tugas ini termasuk pemahaman bahasa alami, pengenalan suara, visi komputer, pengambilan keputusan, dan pemecahan masalah. Kecerdasan Buatan dipelajari di berbagai bidang seperti robotika, pemrosesan gambar, pembelajaran mesin, pengenalan suara, sistem pengenalan suara, dan sistem pakar. Perangkat lunak sistem pakar adalah sebuah program komputer yang dapat memfasilitasi proses kognitif dan pemahaman seseorang untuk menyelesaikan masalah tertentu [9].

## 2.7. API Gemini

API Gemini (*Application Programming Interface Gemini*) merupakan antarmuka pemrograman aplikasi yang dikembangkan oleh Google untuk memanfaatkan kemampuan AI generatif mereka, khususnya model bahasa generatif seperti Gemini. Gemini adalah model kecerdasan buatan yang dilatih untuk menghasilkan teks, memberikan tanggapan alami dan kontekstual dalam percakapan, serta melakukan berbagai tugas berbasis bahasa. LLM Gemini dapat digunakan untuk berbagai keperluan, seperti meringkas teks, menjawab pertanyaan, mengklasifikasikan data, melakukan analisis sentimen, mengekstrak data, membuat content, membuat kode program, dan melakukan percakapan [10]. Saat ini, Gemini Ultra menjadi model LLM pertama yang dapat mengalahkan manusia dalam MMLU (*massive multitask language understanding*), yang menggunakan kombinasi dari 57 bidang ilmu, mulai dari matematika, fisika, sejarah, hukum, pengobatan dan juga etika dalam pengujian baik mengenai pengetahuan umum mau pun kemampuan pemecahan masalah dengan nilai skor mencapai 90%, melampaui skor GPT-4 yang mendapat nilai 86,4% [11].

**3. METODE PENELITIAN**

**3.1. Desain Aplikasi E-commerce dengan Flutter**

Desain aplikasi e-commerce menggunakan Flutter dirancang untuk memberikan pengalaman belanja yang optimal di perangkat mobile dan desktop. Fitur utama yang dibangun berupa:

- a. Navigasi Pengguna: Login, browsing produk, melihat detail produk, dan menambahkan ke keranjang untuk melakukan checkout.
- b. Antarmuka Pengguna (UI): Desain yang responsif dan intuitif.
- c. Integrasi API: Menggunakan *Google Generative AI API Gemini* untuk otomatisasi deskripsi produk.

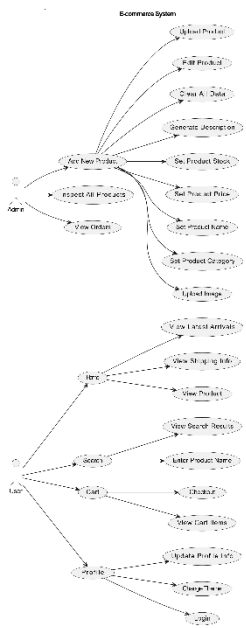
**3.2. Perancangan Sistem**

Perancangan sistem meliputi pembuatan beberapa diagram untuk menggambarkan struktur dan alur kerja aplikasi. Admin dapat menambah, mengedit, dan mengelola produk, termasuk mengunggah gambar, menetapkan kategori, nama, harga, stok, dan menghasilkan deskripsi otomatis menggunakan API Gemini. Admin juga dapat melihat semua produk dan pesanan.

User dapat melihat produk di beranda, mencari produk, melihat informasi pengiriman dan produk terbaru, mengelola keranjang belanja, serta melakukan checkout. User juga dapat mengakses profil untuk login, mengubah tema, dan memperbarui informasi.

**3.2.1. Perancangan Use Case Diagram**

Menurut Sukamto dan Shalahuddin, diagram use case adalah alat pemodelan yang digunakan untuk menggambarkan perilaku sistem informasi yang akan dikembangkan [12].

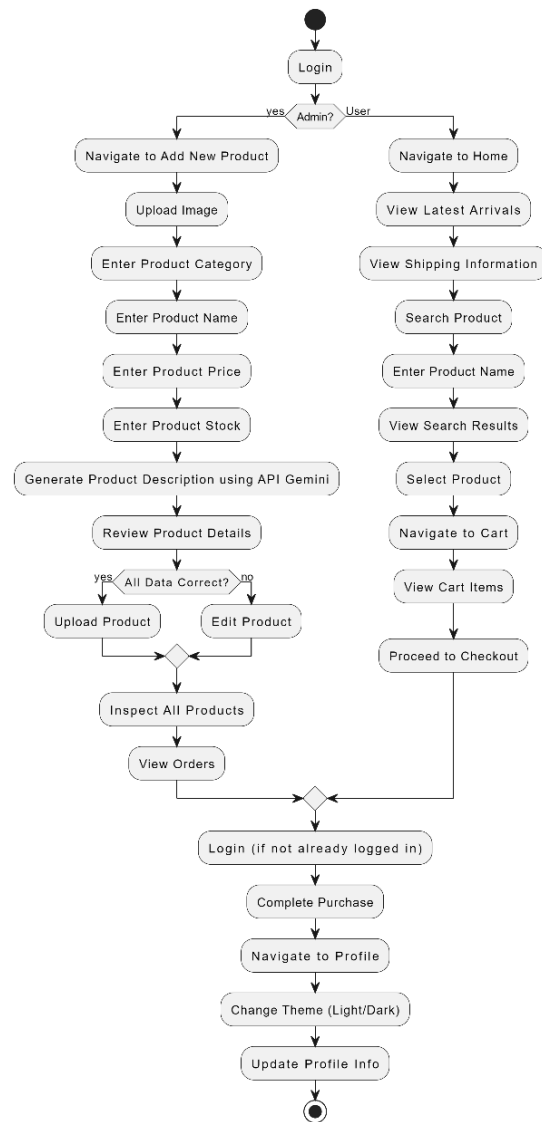


Gambar 1. Use Case

Pada gambar 1 Use Case diagram ini menggambarkan interaksi Admin dan User dengan sistem e-commerce. Diagram ini menunjukkan fungsi-fungsi utama dalam sistem e-commerce dan interaksi kedua aktor tersebut dengan sistem.

**3.2.2. Perancangan Activity Diagram**

Menurut Sukamto dan Shalahuddin, diagram aktivitas menggambarkan aliran kerja atau aktivitas dalam sebuah sistem, proses bisnis, atau menu yang ada pada perangkat lunak [12].



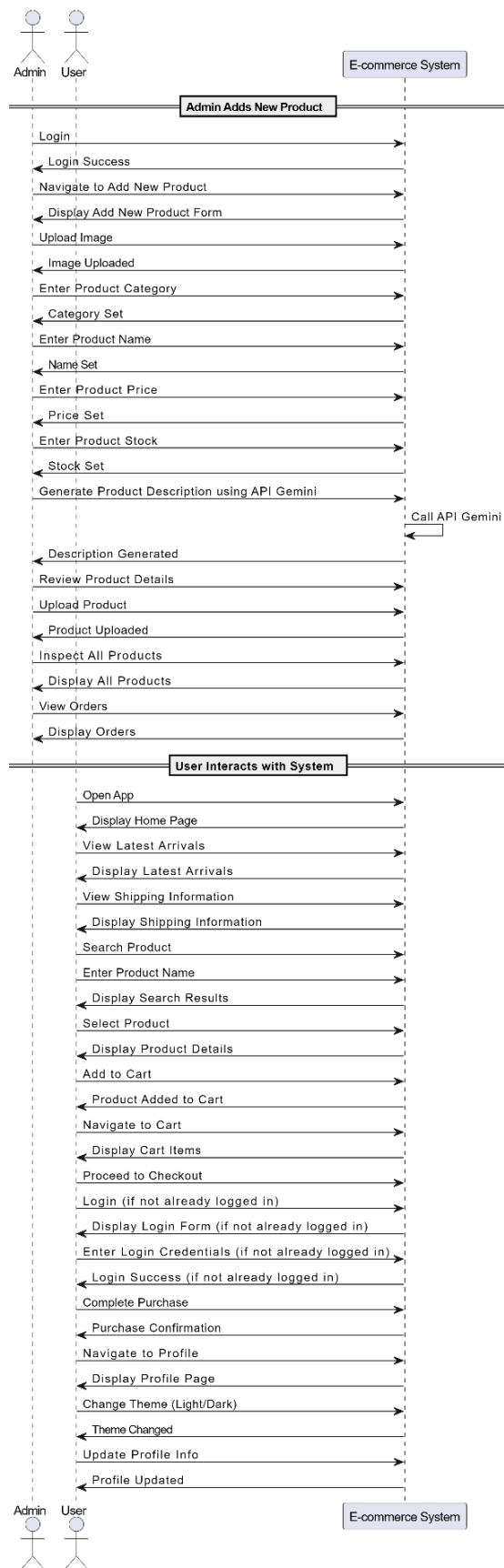
Gambar 2. Activity

Pada gambar 2 activity diagram ini menggambarkan alur kerja Admin dan User dalam sistem e-commerce. Diagram ini menunjukkan langkah-langkah dan keputusan utama yang diambil dalam sistem dari awal hingga akhir.

**3.3. Perancangan Sequence Diagram**

Menurut Sukamto dan Shalahuddin, sequence diagram menampilkan perilaku antar objek dalam sebuah use case dengan melakukan pendekatan pada

masa hidup objek serta pesan-pesan yang dikirim dan diterima antar objek. [12].



Gambar 3. Sequence

Pada gambar 3 *sequence* diagram ini menggambarkan alur interaksi antara Admin dan User dengan sistem e-commerce. Diagram ini memetakan langkah-langkah interaksi antara aktor dan sistem dalam urutan yang logis

### 3.3. Pengembangan Aplikasi

Tahap pengembangan aplikasi mencakup beberapa langkah berikut

#### 3.3.1. Pengaturan Lingkungan Pengembangan

Mengatur lingkungan pengembangan dengan menginstal Flutter SDK, Android Studio, dan Visual Studio Code.

#### 3.3.2. Pembuatan Proyek Flutter

Membuat proyek Flutter baru dan mengonfigurasi struktur proyek.

#### 3.3.3. Implementasi Fitur Utama

Mengimplementasikan fitur-fitur utama seperti autentikasi pengguna, browsing produk, melihat detail produk, menambahkan produk ke keranjang, checkout, serta integrasi dengan API Gemini untuk deskripsi produk.

### 3.4. Integrasi API Gemini

Integrasi API Gemini akan dilakukan dengan menggunakan API untuk mengakses dan memproses informasi dari website e-commerce. Langkah-langkah integrasi meliputi:

- Mengonfigurasi Kunci API: Untuk dapat mengimplementasikan AI pada aplikasi e-commerce ini, maka harus mengatur kunci (*key*) API yang diperlukan untuk mengakses layanan API Gemini. Google telah membuka API Gemini untuk pengembang perangkat lunak dan bisnis yang ingin menggunakan kemampuan Gemini pada aplikasi buatan mereka [13].
- Menghubungkan dengan Backend: Menyusun endpoint yang menghubungkan aplikasi dengan API Gemini.
- Implementasi Deskripsi Produk Otomatis: Menggunakan API untuk menghasilkan deskripsi produk secara otomatis berdasarkan data produk.

### 3.5. Pengujian Aplikasi

Aplikasi e-commerce akan diuji dengan menggunakan metode pengujian integrasi. Pengujian ini dilakukan untuk memastikan bahwa aplikasi dapat berfungsi dengan baik di berbagai platform dan perangkat. Tahapan pengujian meliputi:

- Pengujian Fungsional: Memastikan semua fitur utama berfungsi dengan baik.
- Pengujian Responsif: Memastikan tampilan antarmuka responsif di perangkat pengguna.
- Pengujian Kinerja: Memastikan aplikasi berfungsi dengan optimal termasuk penggunaan penyimpanan perangkat.

- d. Pengujian Integrasi API: Memastikan bahwa integrasi dengan API Gemini pada deskripsi produk berjalan lancar dan menghasilkan deskripsi produk yang akurat.

**4. HASIL DAN PEMBAHASAN**

**4.1. Tampilan Aplikasi Mobile**

Berikut adalah tampilan aplikasi E-commerce Berbasis Mobile menggunakan Flutter dengan fitur fitur deskripsi produk otomatis dengan menggunakan Google Generative AI dengan API Gemini:

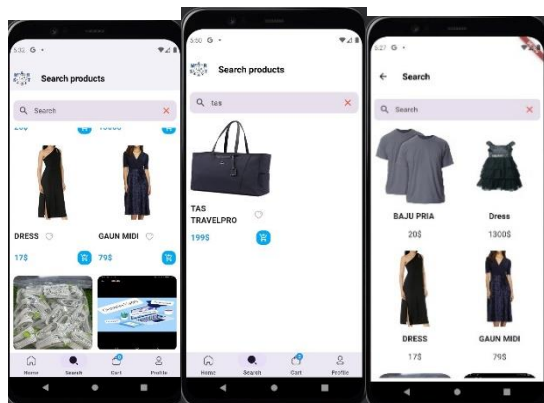
**4.1.1. Tampilan Halaman Home**



Gambar 4. Tampilan menu *home*

Pada gambar 4 adalah halaman menu *home* pada aplikasi E-commerce merupakan yang berisikan logo toko ataupun nama aplikasi yaitu Smart Shop dan berisi daftar kategori produk yang dijual dan keranjang belanja berisi daftar produk yang telah dipilih oleh pengguna untuk dibeli. Pengguna dapat mengklik keranjang belanja untuk melihat detail produk yang ada di dalamnya dan untuk melanjutkan proses checkout.

**4.1.2. Tampilan Halaman Search**

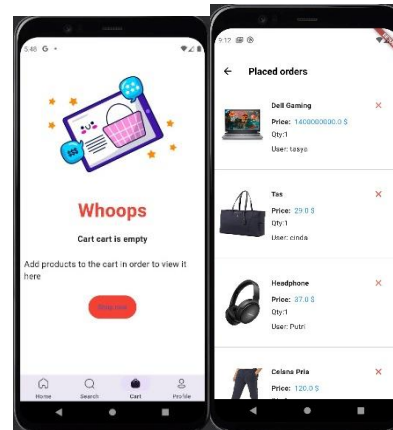


Gambar 5. Tampilan menu *search*

Pada gambar 5 adalah halaman menu pencarian (*Search*) pada aplikasi E-commerce Smart Shop memiliki beberapa fungsi berikut: Pengguna dapat mencari produk dengan mengetikkan nama produk,

dan kategori produk pada bar pencarian dan Pengguna dapat memfilter hasil pencarian berdasarkan nama produk, kategori produk, dan merek produk. Pengguna dapat menambahkan produk ke keranjang belanja dengan mengklik tombol "Tambah ke Keranjang". Pengguna dapat melihat isi keranjang belanja mereka.

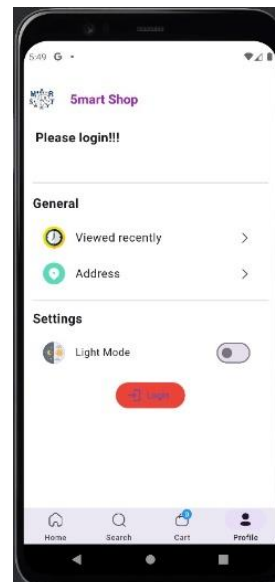
**4.1.3. Tampilan Halaman Cart**



Gambar 6. Tampilan menu *cart*

Pada gambar 6 adalah halaman menu *cart* pada aplikasi E-commerce Smart Shop tersebut, memiliki beberapa fungsi ini termasuk menampilkan daftar produk yang telah dipilih, menampilkan subtotal belanja, menampilkan ongkos kirim, menampilkan user pembeli dan menampilkan tombol "Lanjutkan Belanja".

**4.1.4. Tampilan Halaman Profile**

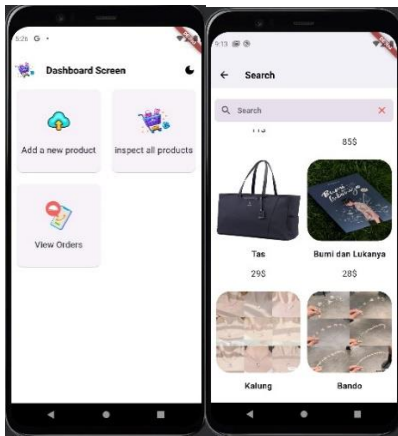


Gambar 7. Tampilan menu *profile*

Pada gambar 7 adalah tampilan menu profile pada aplikasi Smart Shop memiliki banyak fungsi yang membantu pengguna untuk mengelola akun mereka, melihat riwayat pembelian mereka, mengelola

alamat pengiriman mereka, mengelola wishlist mereka, mengubah preferensi mereka, dan keluar dari akun mereka.

**4.1.5. Tampilan Halaman Dashboard Screen**



Gambar 8. Tampilan menu dashboard screen

Pada gambar 8 adalah menu dashboard screen dibuat khusus untuk bagian admin pada aplikasi Smart Shop memiliki fungsi untuk mengelola toko dengan mudah dan efisien. Fungsi-fungsi ini termasuk menambahkan produk baru, melihat daftar produk, melihat daftar pesanan, memproses pesanan, mengelola kategori produk, mengelola pengguna, melihat statistik penjualan, dan mengelola pengaturan toko.

**4.2. Implementasi Fitur API Gemini Berbasis AI**

**4.2.1. Integrasi API Gemini Deskripsi Produk**

Dalam aplikasi 5mart Shop bagian Admin, kami menggunakan API dari Google Generative AI (dikenal sebagai Gemini) untuk menghasilkan deskripsi produk secara otomatis berdasarkan gambar yang diunggah. Berikut adalah penjelasan tentang bagaimana integrasi ini dilakukan dalam program. Pastikan bahwa kunci API yang diperlukan sudah disiapkan yaitu:

“API\_KEY=your\_google\_generative\_ai\_api\_key”  
Kunci API ini digunakan untuk autentikasi saat melakukan permintaan ke API Gemini.

**4.2.2. Mengimpor Pustaka yang Diperlukan**

Kami menggunakan paket flutter\_dotenv untuk memuat variabel lingkungan dari file .env, serta paket google\_generative\_ai untuk berinteraksi dengan API Gemini seperti gambar 9 dibawah ini:

```
import 'dart:convert';
import 'dart:io';

import 'package:flutter_dotenv/flutter_dotenv.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:dotted_border/dotted_border.dart';
import 'package:firebase_storage/firebase_storage.dart';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:image_picker/image_picker.dart';
import 'package:shopsmart_admin_ar/consts/app_constants.dart';
import 'package:shopsmart_admin_ar/models/product_model.dart';
import 'package:shopsmart_admin_ar/services/my_app_method.dart';
import 'package:shopsmart_admin_ar/widgets/subtitle_text.dart';
import 'package:uuid/uuid.dart';
import 'package:http/http.dart' as http;
import 'package:google_generative_ai/google_generative_ai.dart';
```

Gambar 9. Mengimpor kode pustaka yang diperlukan

**4.2.3. Fungsi untuk Menghasilkan Deskripsi**

Kami membuat fungsi \_startGenerateDescription yang akan memanggil API Gemini untuk menghasilkan deskripsi produk berdasarkan gambar yang diunggah:

```
Future<void> _startGenerateDescription() async {
  await dotenv.load(fileName: ".env");
  if (_pickedImage == null) {
    MyAppMethods.showErrorWarningDialog(
      context: context,
      subtitle: "Please pick an image first",
      fct: () {},
    );
    return;
  }
  setState(() {
    _isLoading = true;
  });
  try {
    final apiKey = dotenv.env['API_KEY'];
    if (apiKey == null) {
      print("No API_KEY environment variable");
      MyAppMethods.showErrorWarningDialog(
        context: context,
        subtitle: "API key is missing",
        fct: () {},
      );
      setState(() {
        _isLoading = false;
      });
      return;
    }
    final model = GenerativeModel(model: 'gemini-1.5-pro-latest', apiKey: apiKey);
    final firstImage = await File(_pickedImage.path).readAsBytes(); // Use the picked image
    final prompt = TextPart("Berikan Deskripsi produk untuk di jual supaya menarik");
    final imageParts = [DataPart('image/jpeg', firstImage)];
    final response = await model.generateContent([Content.multi([prompt, ...ImageParts])]);
    final description = response.text ?? '';
    _descriptionController.text = description;
  } catch (error) {
    print("Error generating description: $error");
    MyAppMethods.showErrorWarningDialog(
      context: context,
      subtitle: "An error occurred: $error",
      fct: () {},
    );
  } finally {
    setState(() {
      _isLoading = false;
    });
  }
}
```

Gambar 10. Kode untuk menghasilkan deskripsi

Kami juga menambahkan tombol di UI yang akan memicu fungsi\_startGenerateDescription ketika ditekan:

```
ElevatedButton(
  // Konfigurasi lainnya
  onPressed: _startGenerateDescription,
  child: Text('Generate Description'),
), // ElevatedButton
```

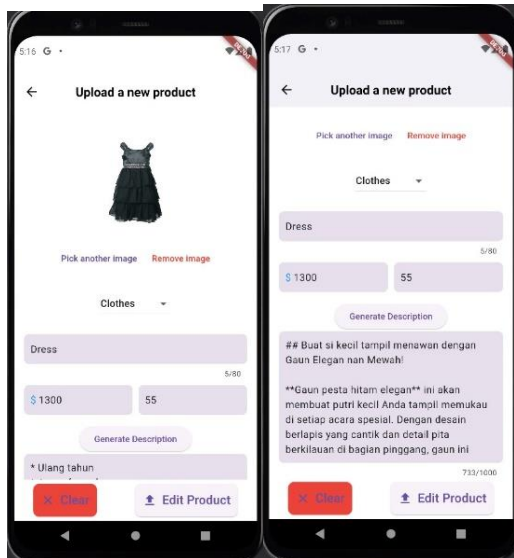
Gambar 11. Kode startGenerateDescription

**4.2.4. Tampilan dan Pengujian Fungsionalitas**

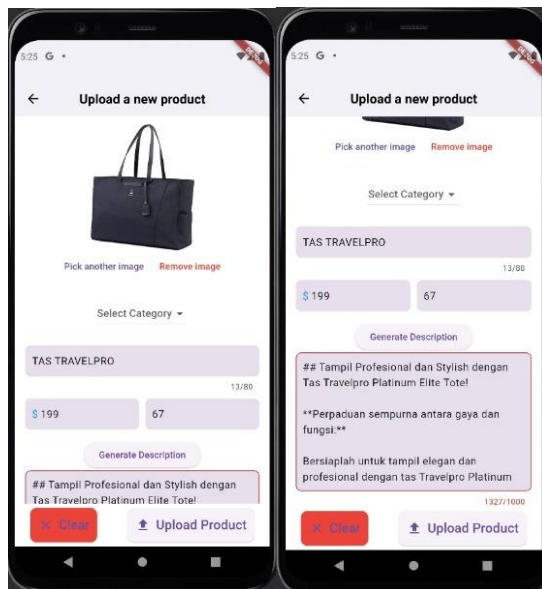
Pada bagian ini, mengimplementasikan fitur deskripsi produk otomatis dengan menggunakan Google Generative AI dengan API Gemini yang diimplementasikan dalam aplikasi E-commerce Smart Shop. Implementasi ini dirancang untuk membantu pengguna dalam berinteraksi dengan aplikasi dan mendapatkan informasi yang mereka butuhkan dengan



cepat dan mudah. Berikut adalah rincian dari tampilan dan pengujian fungsionalitasnya:



Gambar 12. Tampilan fitur deskripsi produk (1)



Gambar 13. Tampilan fitur deskripsi produk (2)

Pada halaman gambar 12 dan gambar 13, pengujian fungsionalitas utama dari fitur tersebut dengan *Google Generative AI* menggunakan kunci API Gemini ini meliputi kemampuan untuk mengunggah dan mengedit produk, termasuk mengelola gambar produk dan menghasilkan deskripsi produk secara otomatis dengan bantuan berbasis AI. Pengguna dapat mengisi form untuk nama produk, harga, kategori, dan jumlah produk, dengan validasi yang memastikan semua data yang diperlukan diisi dengan benar. Gambar produk dapat dipilih, diganti, atau dihapus dengan mudah, dan proses unggahan atau pengeditan produk ditangani dengan mulus dengan feedback visual seperti loading indicator dan notifikasi toast. sedangkan fungsi *Generate Description*

menggunakan model AI untuk menghasilkan deskripsi produk yang menarik.

Dengan fitur ini, pengguna dapat menghasilkan deskripsi produk secara otomatis hanya dengan memilih gambar produk yang ingin diunggah. Prosesnya dijalankan dengan menekan tombol "Generate Description". Setelah gambar dipilih, program akan menggunakan API Gemini untuk menghasilkan deskripsi yang menarik berdasarkan gambar yang dipilih tersebut. Deskripsi yang dihasilkan kemudian ditampilkan di dalam form deskripsi produk, sehingga pengguna dapat mengedit atau menambahkan informasi lain sesuai kebutuhan sebelum mengunggah produknya. Dengan fitur ini, pengguna dapat menghemat waktu dan usaha dalam membuat deskripsi produk yang menarik, sehingga meningkatkan efisiensi dalam pengelolaan toko online mereka. Aplikasi ini memiliki fungsionalitas mengelola dan menyimpan data produk di Firebase, termasuk menyimpan gambar di Firebase Storage untuk mengakses variabel lingkungan, berinteraksi dengan Firebase untuk menyimpan data produk dan gambar, serta pengguna untuk memilih gambar produk dari galeri atau kamera mereka. Selain itu, integrasi dengan *Google Generative AI* menggunakan kunci API Gemini untuk menghasilkan deskripsi produk secara otomatis berdasarkan gambar yang dipilih, yang membantu dalam menciptakan konten yang menarik dan meningkatkan efisiensi.

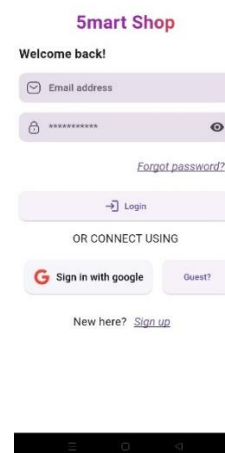
### 4.3. Analisis Sistem

#### 4.3.1. Analisis Fungsional

Pengujian ini dilakukan sebagai seorang pengguna yang menggunakan aplikasi Smart untuk pertama kalinya, yang bertujuan untuk mengevaluasi pengalaman pengguna awal saat menggunakan aplikasi tersebut. Pengujian dilakukan untuk memastikan semua fitur berfungsi sesuai yang diharapkan.

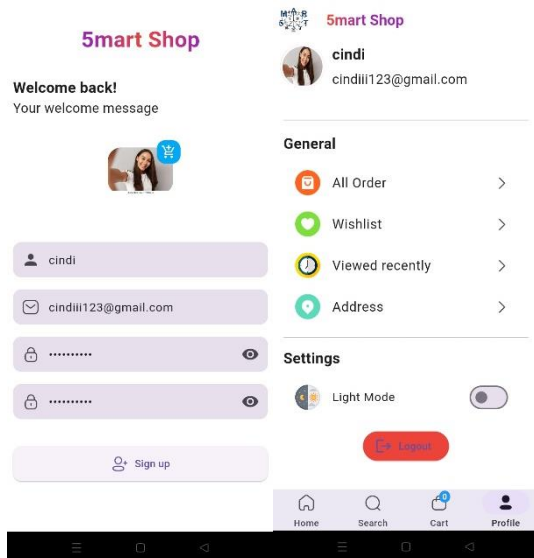
#### 4.3.2. Login dan Registrasi

Proses verifikasi dan autentikasi pengguna berfungsi dengan baik tanpa kendala.

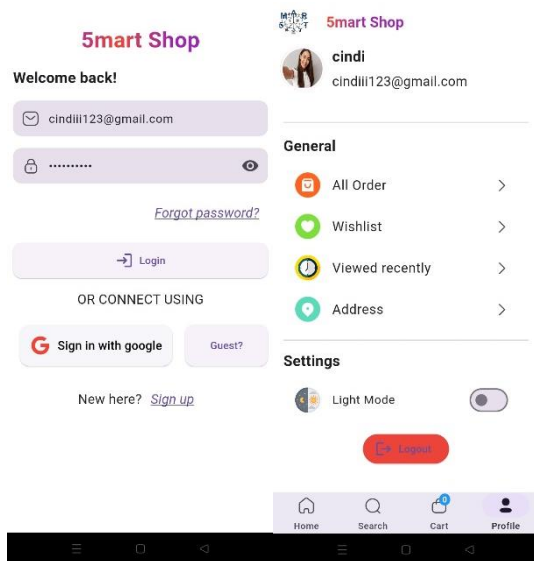


Gambar 14. Fitur Login

Pengguna melakukan register terlebih dahulu jika tidak pernah mendaftar atau memiliki akun pada aplikasi ini sebelumnya. Jika sudah selesai melakukan register, pengguna langsung diarahkan ke halaman awal aplikasi. Pada halaman profil, pengguna dapat melihat tampilan profil sesuai data pada form login atau register yang telah diisi sebelumnya.

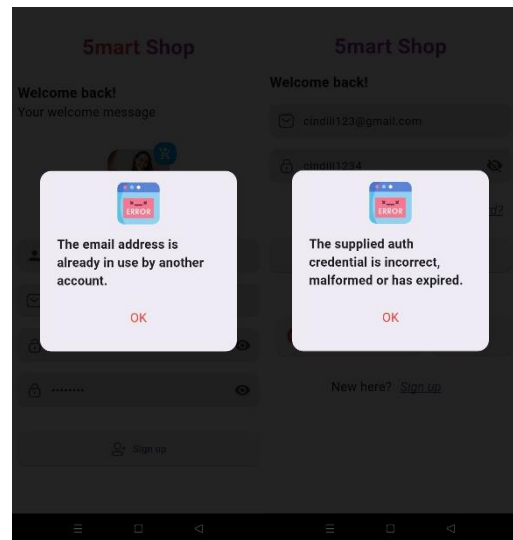


Gambar 15. Fitur Register dan Tampilan Profil



Gambar 16. Lakukan Login Setelah Register

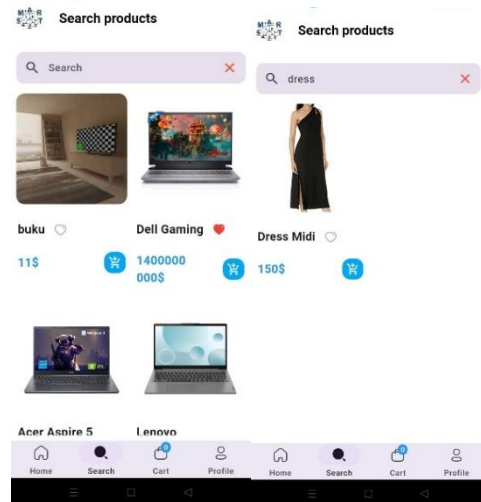
Pengguna diberikan pesan kesalahan yang menunjukkan bahwa alamat email yang Anda masukkan sudah digunakan oleh akun lain. Hal ini terjadi ketika pengguna mencoba untuk mendaftarkan akun baru atau mengubah alamat email pada akun yang sudah ada. Kondisi lainnya yang dapat terjadi yaitu pengguna salah memasukkan halaman email atau password sesuai pengisian yang dilakukan pada halaman register.



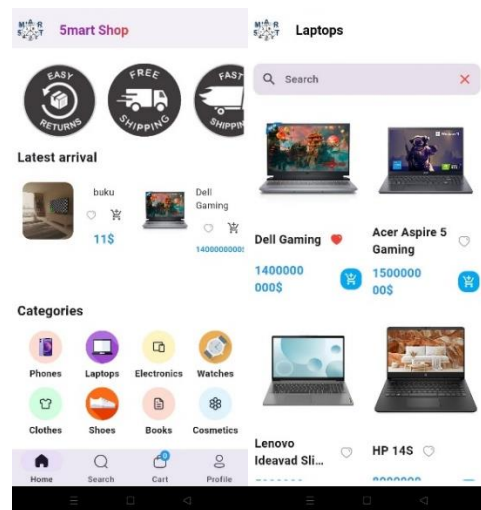
Gambar 17. Pesan Kesalahan

### 4.3.3. Pencarian dan Filter Produk

Fitur pencarian dan filter dapat mengembalikan hasil dengan cepat dan akurat.



Gambar 18. Fitur Pencarian (Search)

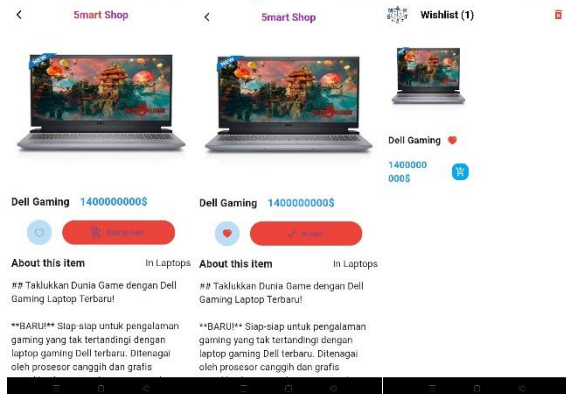


Gambar 19. Fitur Kategori Sebagai Filter Produk

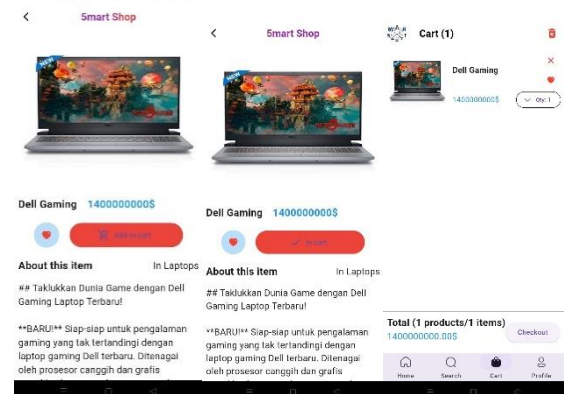


#### 4.3.4. Keranjang Belanja

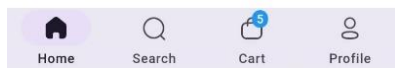
Proses penambahan produk ke keranjang untuk melakukan checkout berjalan lancar. Pada aplikasi ini, pengguna dapat menambahkan produk pada wishlist dengan mengklik tombol suka. Tombol suka yang awalnya Jika ingin menambahkan produk pada cart maka dapat mengklik *add to cart* sehingga aksi tersebut secara cepat merubah tombol menjadi *in cart* yang artinya berhasil menambahkan produk-produk tersebut langsung ke dalam *cart*.



Gambar 20. Menambahkan Produk ke Dalam Wishlist



Gambar 21. Menambahkan Produk ke Dalam Cart



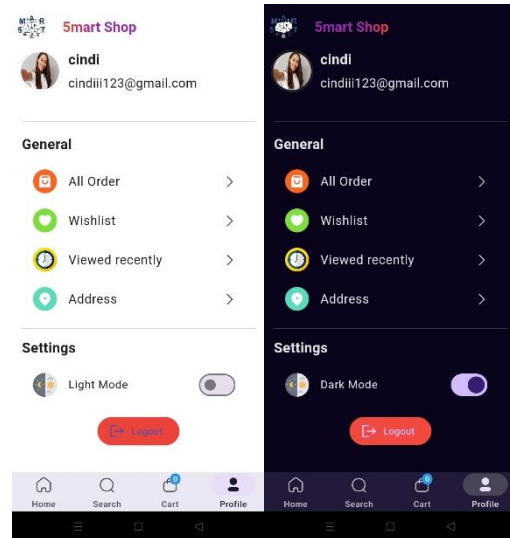
Gambar 22. Penanda Jumlah Produk Pada Cart

#### 4.3.5. Analisis Responsif

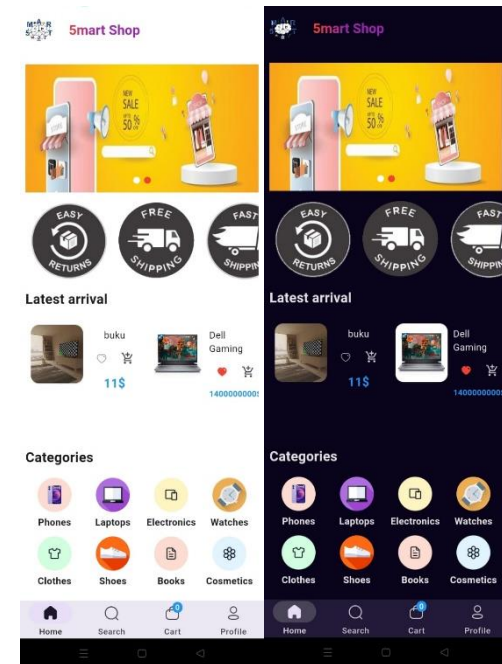
Pengujian responsif bertujuan untuk memastikan bahwa tampilan antarmuka pengguna (UI) aplikasi responsif dan berfungsi dengan baik pada perangkat mobile pengguna. Melakukan pengujian responsif dengan mengubah mode terang dan gelap pada tampilan aplikasi. Proses pengujian mencakup memastikan bahwa teks dan elemen UI tetap dapat dibaca di kedua mode, kontras warna cukup untuk kenyamanan mata, dan tidak ada elemen yang hilang atau terdistorsi ketika beralih antara mode gelap dan terang.

Pada aplikasi Smart Shop ini, peralihan dari mode gelap ke terang terlihat jelas dengan perubahan warna pada font aplikasi, yang awalnya berwarna hitam, menjadi warna putih. Elemen-elemen pada

aplikasi tidak hilang atau terdistorsi serta tidak mengalami perubahan warna sama sekali.



Gambar 23. Mengaktifkan Mode Terang ke Gelap



Gambar 24. Perbandingan Mode Terang dan Gelap

#### 4.3.6. Analisis Kinerja

Pengujian kinerja aplikasi melibatkan pemantauan beberapa aspek, salah satunya adalah penggunaan penyimpanan. Penggunaan penyimpanan mencakup pengukuran jumlah ruang penyimpanan yang digunakan oleh aplikasi setelah instalasi dan pemantauan perubahan penggunaan penyimpanan seiring waktu dan aktivitas aplikasi. Pada awalnya, setelah mengunduh aplikasi, total penggunaan penyimpanan adalah 28,5 MB. Dari jumlah tersebut, 28,5 MB digunakan oleh aplikasi itu sendiri, sedangkan tidak ada penggunaan untuk data dan cache.

Setelah melakukan berbagai aktivitas seperti registrasi, login, mencari produk, serta

menambahkannya ke *wishlist* dan *cart*, penggunaan penyimpanan meningkat menjadi 30,3 MB. Dalam peningkatan ini, ukuran aplikasi tetap 28,5 MB, sementara penggunaan untuk data meningkat menjadi 324 KB dan cache menjadi 1,5 MB.

## 5. KESIMPULAN DAN SARAN

Berdasarkan hasil dan pembahasan diatas dapat disimpulkan aplikasi E-commerce berbasis mobile yang dikembangkan menggunakan *Flutter* mengimplementasikan berbagai fitur yang dapat mempermudah pengguna dalam berbelanja online. Integrasi *API Gemini* dari *Google Generative AI* untuk deskripsi produk otomatis terbukti efektif. Fitur ini agar admin menghasilkan deskripsi produk yang menarik secara otomatis berdasarkan gambar yang diunggah, meningkatkan efisiensi dalam pengelolaan toko online. Pengujian fungsionalitas fitur ini menunjukkan bahwa pengguna dapat dengan mudah mengunggah dan mengedit produk, mengelola gambar produk, dan menghasilkan deskripsi produk berbasis AI dengan validasi form yang memastikan semua data yang diperlukan diisi dengan benar. Integrasi dengan *Firestore* juga penyimpanan data produk dan gambar secara efisien. Analisis sistem menunjukkan bahwa fitur login dan registrasi berfungsi dengan baik tanpa kendala, fitur pencarian dan filter produk bekerja dengan cepat dan akurat, dan proses penambahan produk ke keranjang berjalan lancar.

Pengujian responsif memastikan bahwa UI aplikasi berfungsi dengan baik pada perangkat mobile dalam mode terang dan gelap. Pengujian kinerja aplikasi menunjukkan bahwa penggunaan penyimpanan tetap efisien meskipun melakukan berbagai aktivitas seperti registrasi, login, pencarian produk, dan penambahan produk ke *wishlist* dan *cart*. Untuk penelitian selanjutnya, disarankan untuk mengembangkan lebih lanjut integrasi dengan teknologi AI guna meningkatkan pengalaman pengguna dan efisiensi dalam aplikasi. Selain itu, perlu dilakukan evaluasi lebih lanjut terhadap respons pengguna terhadap fitur-fitur yang telah diimplementasikan untuk meningkatkan kualitas layanan aplikasi. Analisis lanjutan terhadap performa aplikasi juga diperlukan untuk memastikan skalabilitas dan stabilitas yang optimal seiring dengan pertumbuhan jumlah pengguna dan data.

## DAFTAR PUSTAKA

- [1] P. S. P. Aruningtyas, P. Haribowo, dan J. Jumi, "Rancang Bangun E-Commerce Berbasis Web Responsive Dan Mobile Android Pada Umkm Aikori Natural Leather Bag Semarang Sebagai Media Promosi Dan Informasi," *JOBS (Jurnal Of Business Studies)*, vol. 5, no. 1, pp. 73-86, 2019.
- [2] M. Ilham, N. Rahaningsih, and I. Ali, "Aplikasi Mobile E-Commerce Point Coffee pada PT. Indomarco Prismaatama Cabang Cirebon," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 1, pp. 572-578, 2023.
- [3] R. Hermiati, A. Asnawati, and I. Kanedi, "Pembuatan E-Commerce Pada Raja Komputer Menggunakan Bahasa Pemrograman PHP Dan Database MySQL," *Jurnal Media Infotama*, vol. 17, no. 1, 2021.
- [4] A. V. Romero, K. Kusnadi, and R. Fahrudin, "Membangun Marketplace Untuk Penjualan Produk Kreatif Mahasiswa Berbasis Mobile Menggunakan Metode FDD," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 6, pp. 3400-3405, 2023.
- [5] M. F. Osama, I. Purnamasari, and R. Mayasari, "Rancang Bangun Aplikasi Pemesanan Kue Berbasis Multiplatform untuk Mendukung Langkah Ekonomi Digital," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 4, pp. 2877-2883, 2023.
- [6] A. Wahyu, M. Affandes, P. Pizaini, Y. Vitriani, and I. Iskandar, "Aplikasi E-Commerce Galeri Lembaga Adat Melayu Riau Berbasis Mobile Menggunakan Flutter Menerapkan Metode Waterfall," *Journal of Information System Research (JOSH)*, vol. 4, no. 2, pp. 458-469, 2023.
- [7] D. R. Poetra, "Performa Algoritma Bubble Sort dan Quick Sort pada Framework Flutter dan Dart SDK (Studi Kasus Aplikasi E-Commerce)," *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, vol. 9, no. 2, pp. 806-816, 2022.
- [8] B. Raharjo, *Pemrograman Android Dengan Flutter*. Informatika, 2019.
- [9] I. S. Nugroho and A. Voutama, "IMPLEMENTASI CHAT BOT UNTUK PELAYANAN PELANGGAN YANG TERINTEGRASI WEB TOKO KOMPUTER," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 3, pp. 3132-3136, 2024.
- [10] N. Rachmat and D. P. Kesuma, "Implementasi LLM Gemini Pada Pengembangan Aplikasi Chatbot Berbasis Android," *Jurnal Ilmu Komputer (JUIC)*, vol. 4, no. 1, pp. 40-52, 2024.
- [11] S. Pichai dan D. Hassabis, "Introducing Gemini: Our Largest and Most Capable AI Model,". Internet: <https://blog.google/technology/ai/google-gemini-ai/#sundar-note>, Desember 2023 [13 Mei 2024].
- [12] M. Tabrani and H. Priyandaru, "SISTEM INFORMASI MANAJEMEN BERBASIS WEBSITE PADA UNL STUDIO DENGAN MENGGUNAKAN FRAMEWORK CODEIGNITER," *Jurnal Ilmiah M-Progress*, vol. 11, no. 1, pp. 1-9, 2021.
- [13] N. Rachmat and D. P. Kesuma, "Implementasi LLM Gemini Pada Pengembangan Aplikasi Chatbot Berbasis Android," *Jurnal Ilmu Komputer (JUIC)*, vol. 4, no. 1, 202