

# Perancangan Aplikasi Pemesanan Paket Wisata pada Pokdarwis Tangaya Berbasis Android menggunakan Metode Extreme Programming

**Hashbi Alhadi<sup>1</sup>, Arif Amrulloh<sup>2</sup>, Dany Candra Febrianto\*<sup>3</sup>**

<sup>1,3</sup>Program Studi Rekayasa Perangkat Lunak, Universitas Telkom, Kampus Purwokerto

<sup>2</sup>Program Studi Informatika, Universitas Telkom, Kampus Purwokerto

e-mail: <sup>1</sup>hashby@student.telkomuniversity.ac.id, <sup>2</sup>amrulloh@telkomuniversity.ac.id,

<sup>3</sup>danycandra@telkomuniversity.ac.id

\*Penulis Korespondensi

Diterima: 21 Juli 2025; Direvisi: 30 November 2025; Disetujui: 01 Desember 2025

## Abstrak

*Pokdarwis Tangaya di Desa Saniang Baka mengelola berbagai layanan wisata, seperti wisata alam, aktivitas berkemah, dan eduwisata. Proses pemesanan yang masih bergantung pada WhatsApp menyebabkan penumpukan pesan, kesalahan pencatatan, serta potensi penurunan kualitas layanan, terutama pada periode liburan ketika jumlah pengunjung meningkat. Untuk mengatasi permasalahan tersebut, dikembangkan sebuah aplikasi pemesanan berbasis Android menggunakan metodologi Extreme Programming (XP) yang dipilih karena sifatnya yang iteratif, adaptif, dan mendukung perubahan kebutuhan secara cepat. Tahap perancangan sistem dilakukan menggunakan Unified Modelling Language (UML) serta Entity Relationship Diagram (ERD) untuk memodelkan struktur basis data. Aplikasi diimplementasikan menggunakan Dart dan Flutter sebagai kerangka kerja antarmuka, serta Firebase untuk penyimpanan data real-time dan autentikasi pengguna. Sistem ini menyediakan dua level akses, yaitu pengunjung dan admin, sehingga pengelolaan pemesanan, penjadwalan, serta validasi data pengunjung dapat dilakukan secara terstruktur. Hasil pengujian menggunakan metode Black Box Testing dan User Acceptance Testing (UAT) menunjukkan bahwa seluruh fitur dapat berjalan dengan baik dan telah memenuhi kebutuhan pengguna. Dengan demikian, aplikasi ini diharapkan mampu menjadi solusi efektif dalam meningkatkan efisiensi layanan, meminimalkan kesalahan pencatatan, serta mendukung pengelolaan wisata yang lebih profesional.*

**Kata kunci:** *Android, extreme programming, flutter, pokdarwis*

## Abstract

*The Tangaya Tourism Awareness Group in Saniang Baka Village manages nature tourism, camping, and edu-tourism activities. The current booking system relies on WhatsApp, which leads to a backlog of messages and a potential decline in service quality, especially during peak holiday seasons. To address this issue, an application was designed using the Extreme Programming (XP) methodology, selected for its flexibility. The system design phase is depicted with Unified Modeling Language (UML) and an Entity Relationship Diagram (ERD) for database modeling. The implementation utilizes the Dart programming language and the Flutter framework for the user interface, along with Firebase for real-time data management and user authentication. A functional Android application was successfully developed with two access levels: visitor and admin. Black Box Testing and User Acceptance Testing (UAT) demonstrated that all features function as expected and that user requirements have been validly met. This application is expected to serve as an effective solution for managing bookings, scheduling, and ensuring the validity of visitor data.*

**Keywords:** *Android, extreme programming, flutter, pokdarwis*

## 1. PENDAHULUAN

Pokdarwis Tangaya merupakan organisasi masyarakat yang berada di Desa Saniang Baka, Kecamatan X Koto Singkarak, Kabupaten Solok. Pokdarwis Tangaya menyediakan berbagai paket wisata, seperti jelajah alam, perkemahan, eduwisata, serta berbagai kegiatan lainnya yang menjadi bagian dari agenda tahunan Kabupaten Solok. Proses pemesanan paket wisata atau pendaftaran event oleh pengunjung saat ini dilakukan dengan mendatangi sekretariat Pokdarwis Tangaya secara langsung atau melalui WhatsApp.

Proses pemesanan sering mengalami kendala akibat lonjakan pengunjung, terutama pada hari libur. Hal ini menyebabkan penumpukan pesan masuk dan memicu berbagai kesalahan, seperti ketidakcukupan penjadwalan, hilangnya data pengunjung, serta keterlambatan persetujuan pemesanan oleh pengelola. Kondisi tersebut mengganggu kenyamanan pengunjung karena mereka harus menunggu terlalu lama untuk mendapatkan konfirmasi pemesanan, yang pada akhirnya dapat menurunkan minat masyarakat untuk berkunjung ke objek wisata yang dikelola Pokdarwis Tangaya.

Pada era digital saat ini, penggunaan perangkat mobile di Indonesia menunjukkan angka yang sangat tinggi. Data Badan Pusat Statistik (BPS) tahun 2001 hingga 2003 mencatat bahwa lebih dari 60% penduduk Indonesia berusia di atas 5 tahun telah menggunakan perangkat *mobile*. Tingginya tingkat penggunaan tersebut menggambarkan bahwa perangkat *mobile* sudah menjadi media utama dalam mengakses informasi dan layanan digital. Oleh karena itu, pemanfaatan teknologi *mobile* sangat relevan untuk mempermudah pengelolaan data pengunjung pada proses pemesanan tiket, sekaligus memudahkan calon pengunjung dalam memperoleh informasi terkait objek wisata secara cepat, praktis, dan akurat. [1], [2].

Guna mengatasi kendala yang dihadapi oleh Pokdarwis Tangaya maka diusulkan untuk membuat Aplikasi pemesanan tiket wisata berbasis *mobile*. Pengembangan aplikasi ini akan menggunakan metode *Extreme Programming*, yang termasuk salah satu metode *agile*. Metode ini bertujuan untuk mengembangkan perangkat lunak sederhana dan berfokus pada hasil *coding*, metode ini diciptakan oleh Kent Beck, Ron Jeffries dan Ward Cunningham [4],[5]. Metode ini cocok dengan pengembangan aplikasi dengan skala tim yang kecil dan menengah dikarenakan bersifat lebih fleksibel terhadap perubahan sesuai dengan kebutuhan [6]. Berbeda dengan metode *scrum* yang memerlukan tim dengan skala besar karena pada metode *scrum* memiliki prinsip kerja sama tim yang kolaboratif dan iteratif dalam tahap pengembangan [7]. Metode *Extreme Programming* memiliki kerangka kerja dalam pengembangan, seperti *planning* yang digunakan untuk identifikasi masalah dan menganalisa kebutuhan, *design* merupakan tahapan dalam perancangan sistem, perancangan database dan UI/UX, *coding* adalah tahapan penerapan hasil dari *design* dengan menggunakan bahasa pemrograman dan *testing* merupakan tahapan terakhir yaitu pengujian sistem [8]. Dalam pemodelan sistem akan menggunakan *Unified Modelling Language* (UML) yang dapat membantu dalam menvisualisasikan, merancang sistem dan representasi grafis dari struktur sistem [9].

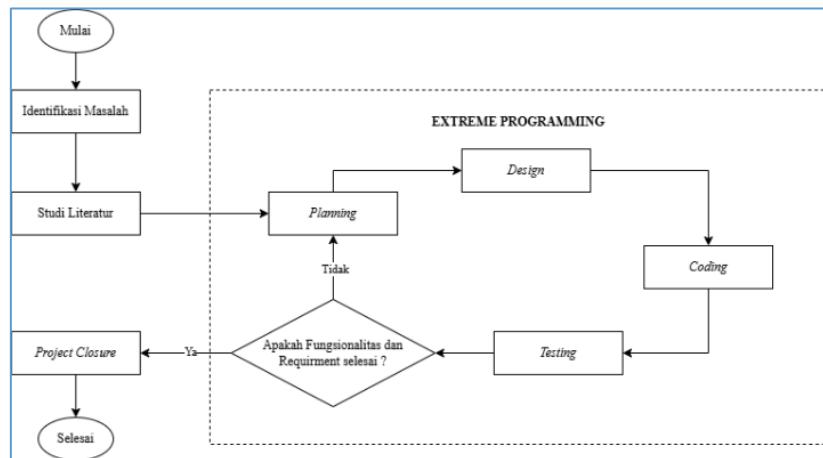
Bahasa pemrograman yang akan digunakan dalam tahapan pengembangan adalah Dart dengan *framework* Flutter yang dikembangkan oleh Google untuk membuat aplikasi *modern* dan sangat populer bagi pengembang karena dilengkapi paket dan fitur yang lengkap sehingga dapat mempercepat dan mempermudah dalam mengembangkan aplikasi *mobile* [10]. Untuk pengelolaan penyimpanan data secara *real-time*, akan menggunakan Firebase yang dikembangkan oleh Google dengan menyediakan berbagai layanan *cloud* yang mempermudah sinkronisasi data secara otomatis [11],[12]. Penelitian ini akan menggunakan pengujian dengan metode *Black Box Testing* yang digunakan untuk mengevaluasi setiap proses dalam sistem dan memastikan apakah fitur aplikasi dapat berjalan sesuai kebutuhan pengguna [13].

Berdasarkan yang telah diuraikan, penelitian ini bertujuan untuk mengembangkan sebuah aplikasi berbasis Android menggunakan metode *Extreme Programming* yang dapat membantu Pokdarwis Tangaya dalam mengelola pemesanan pengunjung sehingga tidak terjadinya kesalahan data, keterlambatan pengunjung dan mengelola data pengunjung sehingga tidak terjadinya kesalahan data pengunjung.

## 2. METODELOGI PENELITIAN

Pelaksanaan penelitian disusun melalui beberapa tahapan utama, yang dimulai dari identifikasi masalah pada sistem pemesanan Pokdarwis Tangaya, kemudian dilanjutkan dengan studi literatur untuk mendukung perancangan solusi.

Proses pengembangan aplikasi menggunakan metode *Extreme Programming* (XP) yang bersifat iteratif hingga seluruh kebutuhan fungsionalitas terpenuhi. *Extreme Programming* (XP) meliputi siklus *planning*, *design*, *coding*, dan *testing*. Keseluruhan alur penelitian, dari tahap awal hingga akhir, dapat dilihat pada gambar 1.



Gambar 1. Diagram alir penelitian.

### 2.1. Identifikasi Masalah

Proses pelaksanaan penelitian ini diawali dengan mengunjungi sekretariat Pokdarwis Tangaya untuk mengidentifikasi masalah yang sedang dihadapi dengan mewawancara koordinator Pokdarwis Tangaya.

### 2.2. Studi Literatur

Studi literatur didefinisikan sebagai tahapan mencari serta menganalisa beberapa literatur yang berkaitan dengan pengembangan sistem informasi pemesanan berbasis Android dari berbagai macam metode. Kemudian dilakukan perbandingan antar metode untuk menentukan metode mana yang tepat untuk digunakan sebagai acuan dalam penelitian.

### 2.3. *Planning*

*Planning* merupakan tahapan awal dari metode *Extreme Programming*. Pada tahapan ini dibuat *User Story* dari hasil wawancara kepada responden. Selanjutnya menentukan kebutuhan fungsional dan non-fungsional yang disesuaikan dengan kebutuhan stakeholder sebagai acuan untuk tahapan *design*.

### 2.4. *Design*

*Design* merupakan tahapan merancang bagaimana suatu sistem akan diimplementasikan untuk memenuhi kebutuhan yang telah didefinisikan pada tahapan *User Story*. Berdasarkan *User Story* yang telah dibuat, selanjutnya akan dilakukan perancangan sistem menggunakan *Unified Modeling Language* dan perancangan struktur data dan hubungan entitas dalam sistem menggunakan *Entity Relationship Diagram*. *Unified Modelling Language* yang akan digunakan:

- Use Case Diagram*, dibuat untuk menggambarkan hubungan antara aktor dengan fungsionalitas utama sistem yang akan dikembangkan.

- B. *Activity Diagram*, dibuat untuk menggambarkan alur kerja atau proses bisnis yang terjadi dalam sistem secara lebih terperinci.
- C. *Class Diagram*, dibuat untuk merepresentasikan struktur sistem, termasuk kelas-kelas, atribut, metode, serta hubungan antar kelas.
- D. *Sequence Diagram*, dibuat untuk menggambarkan interaksi antar objek dalam sistem berdasarkan urutan waktu.

### 2.5. Coding

Proses *coding* dilakukan menggunakan Visual Studio Code sebagai *text editor*. Tahapan ini dimulai dari pembuatan *front end* menggunakan *framework* Flutter dan menggunakan bahasa pemrograman Dart. Sedangkan pada penyimpanan data akan menggunakan Firebase.

### 2.6. Testing

*Testing* aplikasi dilakukan menggunakan metode *Black Box Testing*. Tahapan ini berfokus pada pengujian fungsionalitas semua fungsi fitur yang telah dikembangkan sesuai dengan kebutuhan pengguna. Selain itu, *User Acceptance Testing* (UAT) juga akan dilakukan untuk memastikan bahwa sistem memenuhi harapan dan persyaratan pemangku kepentingan.

### 2.7. Project Closure

Pada tahapan ini, penelitian akan memasuki tahap *project closure* atau penutupan proyek, yang merupakan tahap akhir dari keseluruhan proses perancangan aplikasi.

## 3. HASIL DAN PEMBAHASAN

Bagian menyajikan hasil penelitian yang telah diperoleh melalui proses perancangan, implementasi, dan pengujian sistem, serta pembahasan yang mengaitkannya dengan tujuan penelitian. Analisis dilakukan berdasarkan data yang dikumpulkan selama proses pengembangan serta hasil evaluasi dari pengguna. Pembahasan dalam bab ini bertujuan untuk menjelaskan sejauh mana sistem yang dibangun mampu memenuhi kebutuhan pengguna dan memberikan solusi terhadap permasalahan yang diidentifikasi.

### 3.1. Planning

Tahapan awal dimulai dengan mempelajari proses bisnis dalam Pokdarwis Tangaya berdasarkan hasil wawancara dengan pemangku kepentingan. Tahapan ini memiliki tujuan menentukan kebutuhan sistem seperti kebutuhan fungsionalitas dan kebutuhan non-fungsionalitas.

### 3.2. User Story

*User Story* merupakan bentuk penggambaran kebutuhan dan tujuan pengguna dalam menggunakan aplikasi yang ditulis dalam sudut pandang pengguna. *User Story* dapat dilihat pada tabel 1.

Tabel 1. *User story*

Kode	User Story
SP-01	Sebagai pengunjung, saya ingin <i>login</i> dengan menggunakan akun Google.
SP-02	Sebagai pengunjung, saya ingin melihat paket wisata atau <i>event</i> yang akan dilaksanakan tanpa harus <i>login</i> .
SP-03	Sebagai pengunjung, saya ingin melakukan pemesanan paket wisata atau <i>event</i> .
SP-04	Sebagai pengunjung, saya dapat memilih metode pembayaran melalui <i>online</i> atau bayar di tempat.

Kode	User Story
SP-05	Sebagai pengunjung, saya dapat mengubah data pribadi seperti no telepon, jenis kelamin dan alamat.
SP-06	Sebagai pengunjung, saya dapat melihat informasi pemesanan apakah sudah disetujui atau ditolak.
SA-01	Sebagai admin, saya ingin login dengan akun yang terdaftar atau menggunakan akun Google.
SA-02	Sebagai admin, saya ingin menambah, menghapus dan mengedit paket wisata atau <i>event</i> .
SA-03	Sebagai admin, saya dapat menyetujui atau membatalkan pemesanan pengunjung.
SA-04	Sebagai admin, saya ingin melihat data pemesanan dalam waktu perbulan.

### 3.3. Kebutuhan Fungsional dan Non-fungsional

Tahapan setelah mengetahui *User Story* maka akan diketahui kebutuhan fungsional dan *non-fungsional*. Kebutuhan fungsional adalah spesifikasi yang mendefinisikan bagaimana sistem harus bekerja. Sementara itu, kebutuhan *non-fungsional* berfokus pada aspek kualitas sistem. Kebutuhan fungsionalitas dilihat pada tabel 2. dan kebutuhan *non-fungsionalitas* pada pada tabel 3.

Tabel 2. Kebutuhan Fungsional.

No	Kebutuhan	Deskripsi	Nama Use Case
1	Admin dan Pengunjung dapat <i>login</i> melalui akun Google	Sistem dapat memberikan akses <i>login</i> menggunakan akun Google.	Login
2	Admin dan Pengunjung dapat melihat halaman detail paket wisata dan <i>event</i>	Pengunjung dapat berpindah ke halaman paket wisata dan <i>event</i> yang menampilkan informasi paket wisata dan <i>event</i> secara detail.	Melihat Detail Paket Wisata dan <i>Event</i>
3	Pengunjung yang ingin melakukan pemesanan harus login terlebih dahulu	Pengunjung dapat <i>login</i> dengan akun Google sebelum melakukan pemesanan, selanjutnya pengunjung mengisi form pemesanan dan melakukan pembayaran.	Pemesanan
4	Pengunjung dapat melihat informasi pemesanan	Pengunjung dapat melihat informasi pemesanan apakah sudah disetujui atau belum pada halaman notifikasi.	Melihat Informasi Pemesanan
5	Pengunjung dapat mengedit data pribadi	Pengunjung dapat mengubah data seperti no telpon, jenis kelamin dan Alamat di halaman <i>profile</i> .	Edit <i>Profile</i>
6	Admin dapat mengedit, menambah, menghapus paket wisata	Admin dapat mengedit menambah, menghapus paket wisata dan <i>event</i> pada halaman profil Admin.	Mengelola Paket Wisata dan <i>Event</i>
7	Admin dapat menyetujui atau membatalkan pemesanan pengguna	Admin dapat melihat informasi pemesanan yang dilakukan oleh pengunjung untuk mengonfirmasi atau membatalkan pesanan yang dilakukan oleh pengunjung.	Konfirmasi Pemesanan
8	Admin dapat melihat laporan pemesanan wisata	Admin dapat melihat seluruh data pemesanan wisata dalam waktu perbulan.	Melihat Laporan Pemesanan

Tabel 3. Tabel Non-fungsional.

No	Kebutuhan Nonfungsional	Deskripsi
1	Keamanan data	Sistem harus melindungi data pengguna dengan enkripsi dan mekanisme keamanan.

No	Kebutuhan Nonfungsional	Deskripsi
2	Kompatibilitas	Sistem harus dapat diakses melalui <i>tablet</i> , dan <i>smartphone</i> dengan berbagai ukuran layer.

### 3.4. Design

Tahapan ini menampilkan rancangan dan desain dalam pengembangan sistem aplikasi sesuai dengan kebutuhan yang telah didefinisikan pada *planning*, dengan menggunakan *Unified Modelling Language*. Kebutuhan sistem teridentifikasi, tahap selanjutnya adalah memodelkan interaksi pengguna dengan sistem menggunakan *Use Case Diagram*. Diagram ini memberikan gambaran tingkat tinggi mengenai fungsional utama sistem dan siapa saja aktor yang terlibat.

#### A. Use Case Diagram

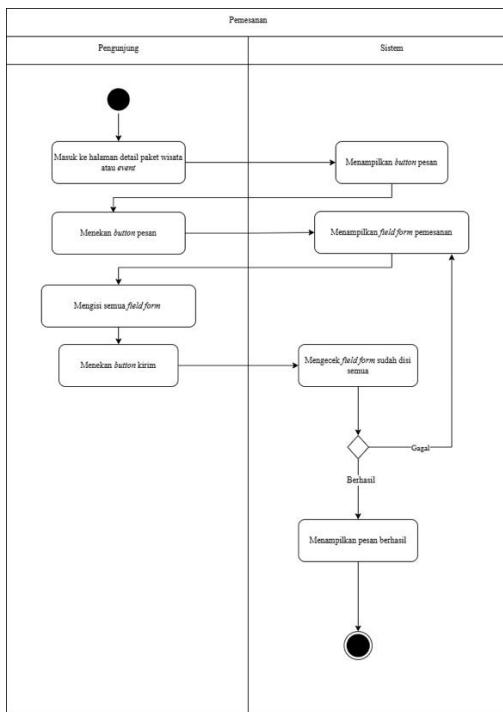
*Use Case Diagram* pada gambar 2 menggambarkan interaksi dan hubungan antara dua aktor, yaitu pengunjung dan admin, dengan sistem. Kedua aktor memiliki peran dan fitur yang berbeda, namun terdapat persamaan dalam hal untuk *login*, dan melihat halaman detail paket wisata serta *event*. Aktor pengunjung dapat melakukan pemesanan paket wisata serta *event* yang akan diselenggarakan dan melihat informasi pemesanan apakah disetujui atau ditolak. Sedangkan untuk aktor admin dapat mengelola paket wisata dan *event*, menerima atau menolak pemesanan yang dilakukan pengunjung dan melihat informasi pemesanan dalam waktu perbulan.



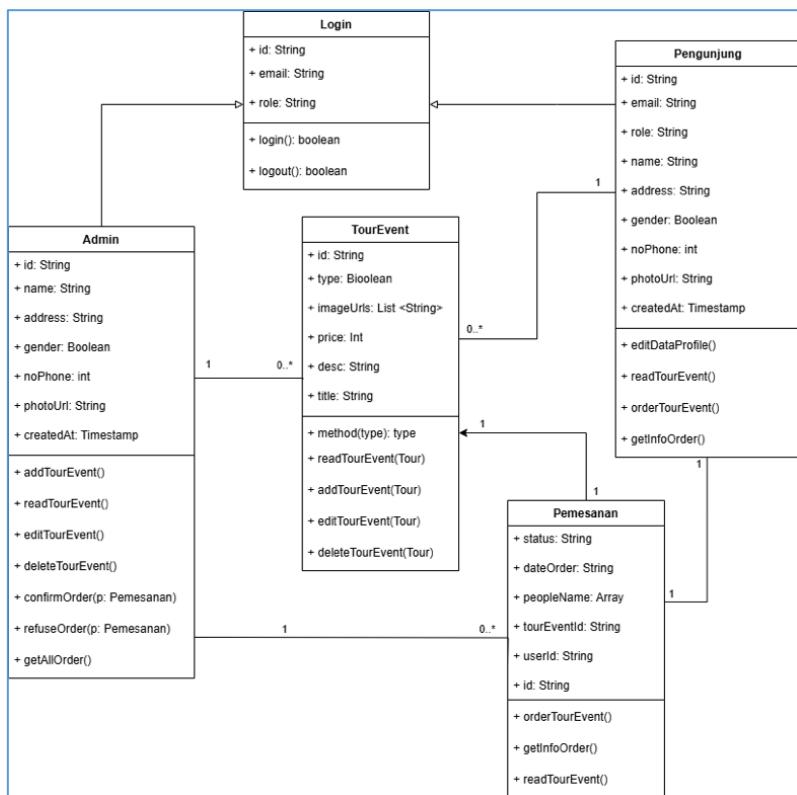
Gambar 2. *Use Case Diagram*

#### B. Activity Diagram

*Activity Diagram* pada gambar 3 menggambarkan proses aktor dalam melakukan pemesanan. Proses diawali dengan aktor berada pada halaman detail paket wisata dan terdapat button pesan. Selanjutnya aktor menekan button pesan dan sistem menampilkan *field form* pemesanan. Aktor mengisi semua *field form* dan setelah mengisi aktor menekan button kirim. Sistem akan mengecek mengecek apakah semua *field* sudah diisi, jika semua *field* masih ada yang kosong akan menampilkan pesan *error* dan jika *field form* sudah diisi semua maka sistem akan mengecek apakah aktor tersebut masih ada pemesanan, jika masih ada pemesanan yang belum diproses maka sistem akan menampilkan pesan *error* dan tidak dapat melakukan pemesanan, jika tidak ada pemesanan, pemesanan berhasil.

Gambar 3. *Activity Diagram* Pemesanan

### C. Class Diagram

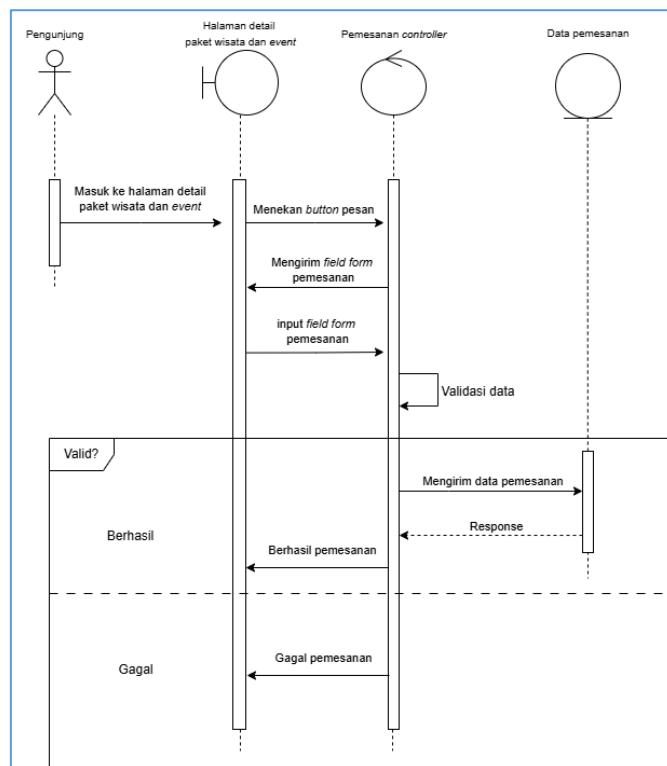
Gambar 4. *Class Diagram*

Class Diagram pada gambar 4 menggambarkan struktur sistem pemesanan paket wisata

berbasis peran pengguna, yaitu admin dan pengunjung, yang masing-masing memiliki relasi dan fungsionalitas terhadap kelas utama seperti *TourEvent* dan Pemesanan. Kelas dasar *Login* menyimpan data autentikasi berupa ‘id’, ‘email’, dan ‘role’, yang kemudian diturunkan pada kelas Admin dan kelas Pengunjung. Keduanya mewarisi identitas dari kelas *Login* dan memiliki atribut serta metode spesifik sesuai peran mereka. Kelas Admin memiliki fungsi untuk menambah, membaca, mengedit, dan menghapus data *TourEvent*, serta mengelola pemesanan dengan metode seperti *confirmOrder()* dan *refuseOrder()*. Sementara kelas Pengunjung dapat melihat daftar *TourEvent*, melakukan pemesanan, dan mengelola profil mereka.

Relasi antar kelas ditunjukkan dengan kardinalitas yang menjelaskan hubungan antar objek. Kelas Admin dapat mengelola banyak *TourEvent* (1 ke 0..\*), dan masing-masing kelas *TourEvent* bisa memiliki banyak kelas Pemesanan (1 ke 0..\*), tetapi setiap kelas Pemesanan hanya mengacu pada satu kelas *TourEvent* dan dibuat oleh satu kelas Pengunjung. Kelas Pengunjung memiliki relasi satu-ke-satu terhadap kelas Pemesanan, yang berarti satu pengguna hanya bisa melakukan satu pemesanan. Diagram ini secara keseluruhan menunjukkan arsitektur terstruktur dari sistem pemesanan wisata berbasis pengguna, mendukung pengelolaan data dinamis oleh admin dan kemudahan akses informasi oleh pengunjung.

#### D. Sequence Diagram



Gambar 5. Sequence Diagram Pemesanan

*Sequence Diagram* pada gambar 5 menggambarkan proses pemesanan paket wisata dan *event*. Diagram ini menggambarkan alur interaksi antara pengunjung, halaman detail paket wisata dan *event*, pemesanan *controller*, dan data pemesanan. Alur dimulai ketika Pengunjung mengakses halaman detail paket wisata dan *event*. Pengunjung kemudian menekan tombol pesan dan pemesanan *controller* mengirimkan field form pemesanan. Pemesanan *controller* menerima input *field form* pemesanan dan melakukan validasi data. Jika validasi berhasil, pemesanan *controller* akan mengirimkan data pemesanan ke data pemesanan. Data pemesanan kemudian memberikan response kembali ke pemesanan *controller* yang menunjukkan bahwa pemesanan

berhasil, dan kemudian pemesanan controller akan mengarahkan kembali ke halaman detail paket wisata dan *event* dengan pesan keberhasilan pemesanan. Sebaliknya, jika validasi gagal, pemesanan *controller* akan mengarahkan kembali ke halaman detail paket wisata dan event dengan pesan kegagalan pemesanan.

### 3.5. Coding

Pada tahap ini dilakukan proses pengembangan sistem melalui penerapan metode Extreme Programming yang berfokus pada kolaborasi, iterasi cepat, dan peningkatan kualitas perangkat lunak.

#### A. Implementasi Awal

Tahap implementasi awal melakukan penyusunan struktur awal sistem agar pengembangan dapat dilakukan secara terorganisir dan mudah dalam pemeliharaan.

##### a. User Interface (Frontend)

*User Interface* aplikasi dikembangkan menggunakan *framework* Flutter dan bahasa pemrograman Dart. Struktur proyek Flutter diorganisir secara modular untuk memastikan setiap komponen dapat digunakan kembali (*reusable*) dan mudah dikelola (*maintainable*).

##### b. Data Management (Backend)

*Data Management backend* untuk sistem ini secara diimplementasikan menggunakan platform Firebase sebagai *Backend-as-a-Service* (BaaS) dan layanan pihak ketiga seperti Midtrans Sandbox dan OpenWeather.

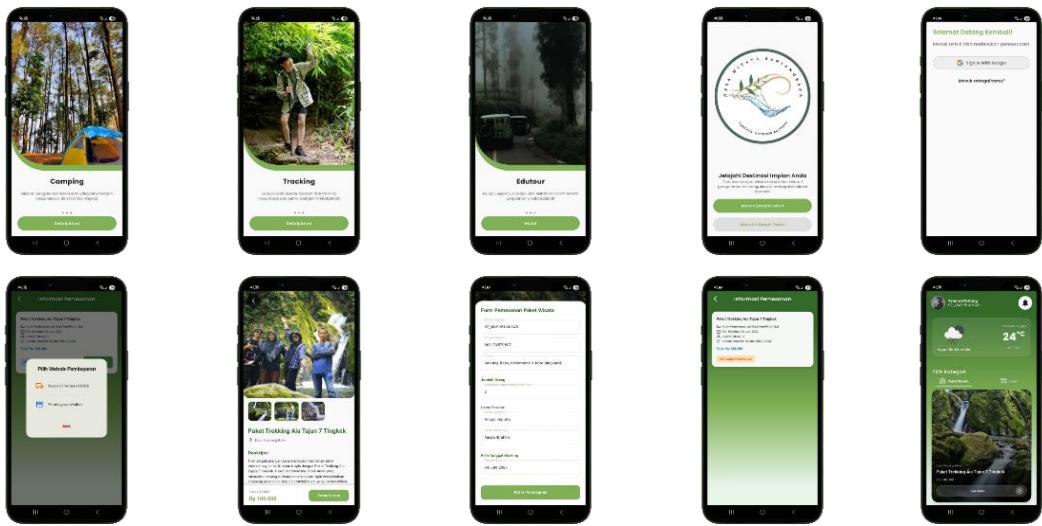


Gambar 6. Arsitektur Backend

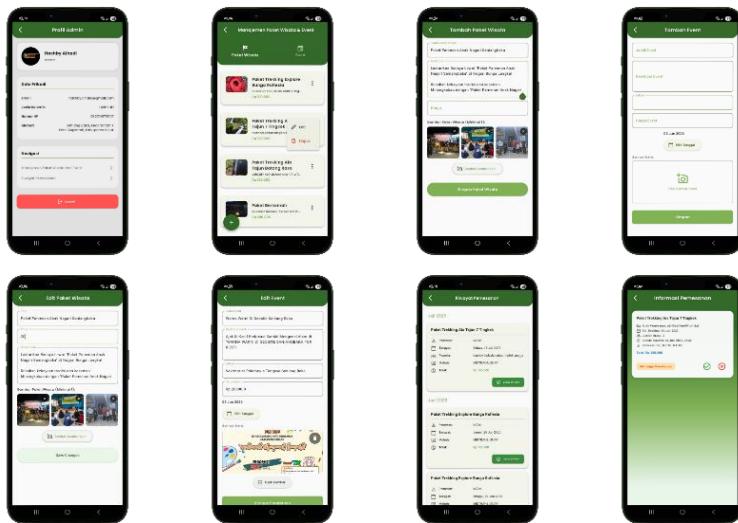
Gambar 6 menggambarkan arsitektur *backend* yang berpusat pada Firebase sebagai backend utamanya, di mana API Services adalah penghubung dengan layanan platform yang memungkinkan untuk melakukan tugas tertentu atau mendapatkan data spesifik. Pengguna masuk melalui Firebase Authentication, dan menyimpan data pada Firestore dengan *collection users*. Penyimpanan dan mengambil data seperti file gambar pada Storage. Proses pembayaran, pertama mengambil detail harga dan nama pemesan dari koleksi di Firestore pada *collection orders* sesuai pesanan, lalu menggunakan untuk membuat simulasi transaksi pembayaran di Midtrans Sandbox. OpenWeather untuk mendapatkan data cuaca terkini.

#### B. Implementasi Akhir

Tahap Implementasi Akhir merupakan hasil dari keseluruhan proses pengembangan sistem. Berikut dalam gambar 7 dan gambar 8 merupakan tampilan antarmuka pengguna yang telah berhasil diimplementasikan.



Gambar 7. Tampilan Pengunjung



Gambar 8. Tampilan Admin

### C. Refactor Code

Refactoring dilakukan setelah fitur utama sistem berhasil diimplementasikan dan berjalan sesuai kebutuhan. Refactor ini bertujuan untuk meningkatkan kualitas kode tanpa mengubah fungsionalitasnya. Berikut adalah beberapa aktivitas refactor yang dilakukan dalam perancangan aplikasi Pokdarwis Tangaya.

#### 3.4. Testing

Pada tahap ini, memperlihatkan hasil pengujian fungsionalitas aplikasi berdasarkan kebutuhan pengguna. Pengujian dilakukan dengan menggunakan metode *Black Box Testing* dan *User Acceptance Testing* (UAT).

##### A. Black Box Testing

*Black Box Testing* dilakukan untuk mengevaluasi fungsionalitas aplikasi berdasarkan spesifikasi kebutuhan, tanpa memperhatikan struktur kode internalnya. Pengujian ini mencakup skenario penggunaan untuk dua jenis aktor, yaitu pengunjung dan admin, di mana hasil lengkapnya dapat dilihat pada tabel 4.

Tabel 4. Black Box Testing

No.	Kode	Deskripsi Pengujian	Hasil Yang Diharapkan	Status
1	SP-01 dan SA-01	Pengunjung dan Admin: login menggunakan akun Google	Menampilkan halaman <i>home</i>	Valid
2	SP-02 dan SA-02	Pengunjung dan Admin: melihat halaman detail paket wisata dan <i>event</i>	Menampilkan halaman detail paket wisata dan <i>event</i>	Valid
3	SP-03	Pengunjung: melakukan pemesanan paket wisata atau <i>event</i>	Berhasil mengirimkan pemesanan paket wisata atau <i>event</i>	Valid
4	SP-04	Pengunjung: mengubah data pribadi	Berhasil mengubah data pribadi	Valid
5	SP-05	Pengunjung: melihat informasi status mengenai pemesanan dan pembayaran	Berhasil melihat informasi mengenai pemesanan dan pembayaran	Valid
6	SA-03	Admin: Mengelola data paket wisata dan <i>event</i>	Berhasil menambah, mengedit atau menghapus paket wisata dan <i>event</i>	Valid
7	SA-04	Admin: Menyetujui atau menolak pemesanan paket wisata dan <i>event</i> yang dilakukan pengunjung	Berhasil menyetujui atau menolak pemesanan pengunjung	Valid
8	SA-05	Admin: Melihat laporan bulanan pemesanan	Berhasil melihat laporan bulanan pemesanan	Valid

Untuk menghitung persentase keberhasilan dari pengujian fungsionalitas yang telah dilakukan, digunakan rumus pada persamaan berikut:

$$\text{Persentase Keberhasilan} = \frac{\text{Jumlah fitur yang berhasil diuji}}{\text{Total fitur yang diuji}} \times 100 \quad (1)$$

$$\text{Presentase Keberhasilan} = \frac{8}{8} \times 100 \quad (2)$$

$$\text{Presentase Keberhasilan} = 100 \quad (3)$$

Berdasarkan hasil pengujian menggunakan metode *Black Box Testing*, seluruh fitur utama dalam telah menunjukkan keluaran yang sesuai dengan spesifikasi yang telah ditentukan.

#### B. *User Acceptance Testing* (UAT)

*User Acceptance Testing* (UAT) dilakukan untuk memvalidasi bahwa aplikasi telah memenuhi kebutuhan dan dapat diterima oleh Pokdarwis Tangaya. Pengujian ini dilakukan langsung oleh Koordinator Pokdarwis Tangaya dan 5 staff pengelola Pokdarwis Tangaya, dengan mengirimkan file aplikasi versi *build* (.apk) beserta skenario pengujian, dan diminta untuk memberikan umpan balik melalui Google Forms yang telah disiapkan. Berikut adalah pertanyaan yang diberikan kepada *responden* pada Tabel 5.

Tabel 5. Tabel Pertanyaan

No.	Pertanyaan	Kode
1	Apakah proses pemesanan kunjungan berjalan lancar dari awal hingga akhir?	UA-01
2	Apakah proses pemesanan hingga pembayaran mudah dipahami?	UA-02
3	Apakah Anda bisa dengan mudah menyetujui atau menolak permintaan kunjungan?	UA-03
4	Apakah fitur pengelolaan jadwal kunjungan membantu mempermudah koordinasi kunjungan wisata?	UA-04
5	Apakah data pengunjung yang masuk ke sistem tercatat dengan baik dan lengkap?	UA-05
6	Apakah setiap fitur mengelola paket wisata atau event mudah digunakan seperti (menambah data, menghapus data dan mengedit data)	UA-06

No.	Pertanyaan	Kode
7	Apakah tampilan aplikasi ini mudah dipahami dan digunakan oleh anggota Pokdarwis?	UA-07
8	Apakah aplikasi tangaya_apps sudah memenuhi kebutuhan dan harapan dari Pokdarwis Tangaya	UA-08

Data yang sudah didapatkan dari *responden*, kemudian dihitung nilai rata-rata dengan cara jumlah jawaban dikali dengan bobot penilaian. Hasil perhitungan rata-rata setiap jawaban pada Tabel 6.

Tabel 6. Persentase Nilai Rata-rata

Kode	Mean	Persentase (%)	Nilai Rata Rata (%)
UA-01	$28/6 = 4.6$	$4.6/5 \times 100 = 93$	
UA-02	$26/6 = 4.3$	$4.3/5 \times 100 = 86$	
UA-03	$29/6 = 4.8$	$4.5/5 \times 100 = 96$	
UA-04	$29/6 = 4.8$	$4.5/5 \times 100 = 96$	
UA-05	$28/6 = 4.6$	$4.6/5 \times 100 = 93$	94%
UA-06	$29/6 = 4.8$	$4.5/5 \times 100 = 96$	
UA-07	$29/6 = 4.8$	$4.5/5 \times 100 = 96$	
UA-08	$29/6 = 4.8$	$4.5/5 \times 100 = 96$	

Berdasarkan hasil persentase nilai rata-rata dari semua pertanyaan pada Tabel 6 didapatkan 94% menyatakan aplikasi memenuhi kebutuhan dan harapan Pokdarwis Tangaya.

### 3.5. Project Closure

Pada tahapan ini, penelitian memasuki tahap *project closure* atau penutupan proyek, yang merupakan tahap akhir dari keseluruhan proses perancangan aplikasi Pokdarwis Tangaya dengan metode *Extreme Programming*. Kegiatan dalam tahap ini termasuk dokumentasi sistem secara lengkap (buku panduan penggunaan, *source code* aplikasi, dan alur kerja sistem), serta penyerahan aplikasi dalam versi *release* (.apk) kepada pengelola Pokdarwis Tangaya.

## 4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan dengan mengimplementasi metode *Extreme Programming* telah berhasil merancang sebuah sistem aplikasi sesuai dengan seluruh fungsionalitas yang direncanakan, baik untuk sisi pengguna sebagai pengunjung maupun admin. Keberhasilan implementasi fitur-fitur tersebut didukung oleh pengujian *Black Box Testing* dan *User Acceptance Testing* yang menunjukkan bahwa seluruh fungsionalitas berjalan sesuai dengan hasil yang diharapkan dan dinyatakan valid, sehingga sistem ini telah memenuhi aspek fungsionalitas yang ditetapkan. Dengan demikian, aplikasi yang dibangun telah mencapai tujuan utamanya dalam merancang aplikasi Pokdarwis Tangaya berbasis Android yang dapat mengelola pemesanan pengunjung dalam menentukan jadwal kunjungan, keterlambatan persetujuan dan data pengunjung Pokdarwis Tangaya.

## 5. SARAN

Meskipun sistem aplikasi yang dikembangkan telah berhasil memenuhi kebutuhan fungsional yang ditetapkan, terdapat beberapa aspek yang dapat dipertimbangkan untuk pengembangan lebih lanjut guna meningkatkan kualitas dan cakupan layanan. Adapun saran-saran yang dapat diberikan yaitu dengan mengoptimasi kinerja dan keamanan berkala yaitu melakukan evaluasi dan optimasi kinerja aplikasi secara berkala, serta melakukan pembaruan keamanan untuk melindungi data pengguna dan menjaga keandalan sistem seiring dengan

perkembangan teknologi.

Menambah fitur notifikasi real-time yang lebih lanjut, selain notifikasi status pemesanan di dalam aplikasi, dapat dikembangkan notifikasi push ke perangkat pengguna atau notifikasi melalui email untuk pembaruan penting, seperti pengingat event, konfirmasi pemesanan, atau perubahan jadwal. Kemudian penambahan fitur ulasan dan peringkat, dengan mempertimbangkan penambahan fitur bagi pengunjung untuk memberikan ulasan dan peringkat terhadap paket wisata atau *event* yang telah diikuti. Fitur ini dapat menjadi masukan berharga bagi Pokdarwis Tangaya dan membantu calon pengunjung lain dalam mengambil keputusan.

Dengan adanya pengembangan berkelanjutan berdasarkan saran-saran tersebut, diharapkan aplikasi Pokdarwis Tangaya dapat semakin berkembang, memberikan manfaat yang lebih besar, dan meningkatkan pengalaman pengguna secara keseluruhan.

## DAFTAR PUSTAKA

- [1] I. Fakhruddin, N. Rismawati, and R. Sriyanti, “*Perancangan Sistem Aplikasi Penjualan Paket Wisata Pulau Seribu Berbasis Java Dengan Metode Scrum (Studi Kasus PT. Abarter Global Indonesia)*,” *J. Nas. Komputasi dan Teknol. Inf.*, vol. 5, no. 1, pp. 104–111, 2022.
- [2] D. B. Pamungkas and A. Suhendar, “*Perancangan Aplikasi Pemesanan Tiket Pariwisata Berbasis Mobile di Kota Yogyakarta*,” *G-Tech J. Teknol. Terap.*, vol. 8, no. 1, pp. 713–721, 2024.
- [3] M. Ramadhan and A. Gustalika, “*Rancang Bangun Aplikasi Pemesanan Tiket Tempat Wisata Berbasis Android Menggunakan Metode Extreme Programming*,” *Bull. Inf. Technol.*, vol. 5, no. 2, pp. 114–124, 2024.
- [4] M. Farhan, K. Darussalam, R. C. Saphira, and F. Purwani, “*Implementasi Extreme Programming Dalam Pengembangan Aplikasi Mobile Pengenalan Pada Masa Orientasi Mahasiswa*,” vol. 14, no. 2, pp. 80–149, 2024.
- [5] T. Ardiansah, “*Perancangan Sistem Persediaan Menggunakan Metode Extreme Programming*,” *J. Ilm. Inform. dan Ilmu Komput.*, vol. 1, no. 1, pp. 1–6, 2022.
- [6] M. Alda, “*Rancang Bangun Sistem Informasi Company Profile Berbasis Android Menggunakan Metode Extreme Programming (XP)*,” *JIKO (Jurnal Inform. dan Komputer)*, vol. 7, no. 1, pp. 62–71, 2023, doi: 10.26798/jiko.v7i1.701.
- [7] A. Mustika, “*Permodelan Sistem Informasi Penjualan Barang Menggunakan Metode Scrum*,” *J. Data Sci. Inf. Syst.*, vol. 2, no. 1, pp. 1–8, 2024.
- [8] M. Ridwan Nawawi, S. Lestanti, and D. Fanny, “*Rancang Bangun Sistem Informasi Inventaris Fasilitas Pondok Pesantren Nurul Ulum Dengan Menggunakan Metode Xp (Extreme Programming)*,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 6, no. 2, pp. 835–841, 2022.
- [9] G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modelling Language User Guide*. 1999.
- [10] A. H. Rio Risqi and Z. Zukhri, “*Penggunaan Firebase Analytics pada Pengembangan Aplikasi Mobile I'm UII dengan Framework Flutter*,” *Automata*, vol. 4, pp. 1–7, 2023.
- [11] S. Kasus, M. Al, R. A. Nugraha, M. T. Ananta, and A. H. Brata, “*Pengembangan Aplikasi Perangkat Bergerak Sistem Manajemen Pembelian dan Patungan Hewan Kurban Berbasis Android*,” vol. 1, no. 1, pp. 1–10, 2017.
- [12] A. A. Shonta, L. N. Hamidah, M. Hasan, M. M. Dewi, Y. Astuti, and I. R. Wulandari, “*Penerapan Firebase Realtime Database Pada Aplikasi Media Informasi dan Pendaftaran Training IT Berbasis Android*,” *J. Media Inform. Budidarma*, vol. 6, no. 3, p. 1517, 2022.
- [13] E. U. Albab, C. D. Suhendra, and L. F. Marini, “*Implementasi Metode Prototype dalam Pengembangan Aplikasi Wondama-Tourism Berbasis Android*,” *Al Qalam J. Ilm. Keagamaan dan Kemasyarakatan*, vol. 18, no. 2, p. 1413, 2024.