



Pengembangan Aplikasi *Mobile Gis* Berbasis *Android* Dengan Integrasi *Google Maps* Dan Studi Perbandingan Pemrograman *Dart/Flutter*

Ahmad Pajar Bahri

Prodi Teknik Elektro, Fakultas Teknik dan Ilmu Komputer, Universitas Islam DR Khez

Muttaqien, Indonesia

Email : Phajarbahri@gmail.com

Abstract

The increase in community mobility has triggered a need for accurate and easily accessible digital map information. This research aims to develop a Mobile GIS application based on Android integrated with Google Maps to facilitate users in determining travel routes. This application utilizes GPS, LBS (Location Based Service), and Google Maps API technologies to provide real-time location and route information. Furthermore, a comparative study with modern programming using Dart/Flutter as an alternative for mobile application development was also conducted. The results show that the built application can run well on Android devices with a minimum specification of the Gingerbread OS.

Keywords: *Mobile GIS, Google Maps, Android, Dart/Flutter.*

Abstrak

Meningkatnya mobilitas masyarakat memicu kebutuhan akan informasi peta digital yang akurat dan mudah diakses. Penelitian ini bertujuan untuk mengembangkan aplikasi Mobile GIS berbasis Android yang terintegrasi dengan Google Maps untuk memudahkan pengguna dalam menentukan rute perjalanan. Aplikasi ini memanfaatkan teknologi GPS, LBS (Location Based Service), dan Google Maps API untuk memberikan informasi lokasi dan rute secara real-time. Selain itu, dilakukan pula studi perbandingan dengan pemrograman modern menggunakan Dart/Flutter sebagai alternatif pengembangan aplikasi mobile. Hasil penelitian menunjukkan bahwa aplikasi yang dibangun dapat berjalan dengan baik pada perangkat Android dengan spesifikasi minimal OS Gingerbread.

Kata Kunci: *Mobile GIS, Google Maps, Dart/Flutter.*

PENDAHULUAN

Perkembangan teknologi smartphone telah membawa perubahan signifikan dalam kehidupan sehari-hari, terutama dalam hal mobilitas. Penggunaan peta digital menjadi kebutuhan utama bagi masyarakat yang sering bepergian ke tempat baru. Namun, seringkali pengguna kesulitan mengingat rute atau lokasi tertentu. Oleh karena itu, diperlukan aplikasi yang dapat memberikan informasi rute secara akurat dan mudah diakses (Riyanto, 2010).

Penelitian ini menggabungkan konsep *Mobile GIS* dengan teknologi *Google Maps* untuk membuat aplikasi penunjuk rute kendaraan pribadi. Selain itu, dilakukan eksplorasi pengembangan aplikasi menggunakan *Flutter* sebagai perbandingan dengan pengembangan *native Android* menggunakan *Java*.

LANDASAN TEORI

1. Sistem Informasi Geografis (GIS)

GIS adalah sistem berbasis komputer yang digunakan untuk mengelola, menganalisis, dan memvisualisasikan data geografis.

2. Mobile GIS

Integrasi perangkat lunak dan keras untuk mengakses data geospasial melalui perangkat bergerak.

3. GPS dan LBS

GPS digunakan untuk menentukan posisi pengguna, sedangkan LBS menyediakan informasi berdasarkan lokasi geografis.

4. Android dan Flutter

- **Android:** Sistem operasi mobile berbasis *Linux* yang *open-source*.
- **Flutter:** *Framework UI* dari *Google* untuk membangun aplikasi *mobile* menggunakan bahasa *Dart*.

METODOLOGI PENELITIAN

A. Alat dan Bahan

- Laptop ACER 4745G
- *Smartphone* Lenovo S930
- *Software*: ADT Bundle, Eclipse, Google Maps API, Flutter SDK

B. Pengembangan Aplikasi

1. Aplikasi Android Native (Java)

- Integrasi *Google Maps API*
- Pembuatan *UI* dengan *XML*
- Implementasi *GPS* dan *LBS*

2. Aplikasi dengan Flutter (Dart)

- Pembuatan *UI* dengan *widget Flutter*
- Integrasi *Google Maps* dengan *plugin*
- Pengujian pada *emulator* dan perangkat fisik

C. Tahapan Pengembangan

1. Perancangan sistem
2. Pengkodean
3. Pengujian
4. Implementasi

HASIL DAN PEMBAHASAN

1. Aplikasi Android Native

Aplikasi bernama PwkTrack telah berhasil dibangun dan diimplementasikan sebagai solusi nyata dari penelitian ini. Aplikasi ini berfungsi sebagai pemandu navigasi yang menampilkan peta *digital* secara interaktif melalui integrasi langsung dengan layanan *Google*

Maps. Pengguna tidak hanya dapat melihat peta statis, tetapi juga berinteraksi dengannya, seperti melakukan *zoom in* dan *zoom out* untuk melihat detail area tertentu. Keunggulan utama dari aplikasi ini adalah kemampuannya dalam menentukan rute perjalanan untuk kendaraan pribadi. Sistem akan secara otomatis menghitung dan menampilkan jalur teroptimal dari posisi pengguna saat ini, yang dideteksi via *GPS*, menuju ke lokasi tujuan yang dipilih. Rute ini ditampilkan secara visual di atas peta sebagai garis berwarna, disertai dengan informasi perkiraan jarak dan waktu tempuh, sehingga memudahkan pengguna dalam melakukan perencanaan perjalanan.

Selain fitur navigasi yang real-time, PwkTrack juga dilengkapi dengan fungsi manajemen data lokasi. Pengguna memiliki kemampuan untuk menyimpan titik-titik lokasi tertentu yang dianggap penting, seperti lokasi rumah, kantor, atau tempat-tempat favorit, ke dalam daftar penyimpanan internal aplikasi. Fitur ini sangat berguna untuk mengingat lokasi-lokasi yang sering dikunjungi tanpa harus mencarinya berulang kali. Untuk menjaga keamanan data dari kehilangan, aplikasi ini juga dilengkapi dengan fungsi backup dan restore. Data lokasi yang telah disimpan dapat diekspor (*dibackup*) ke dalam sebuah *file* yang disimpan di memori eksternal perangkat. *File backup* ini nantinya dapat diimpor kembali (*restore*) ke dalam aplikasi, baik pada perangkat yang sama setelah dilakukan *reset* ataupun ketika pengguna mengganti perangkat *Android* baru, sehingga daftar lokasi penting mereka tetap terjaga dan tidak hilang.

Dari sisi teknis, implementasi fitur-fitur tersebut melibatkan beberapa komponen kunci. Integrasi dengan *Google Maps API* memungkinkan peta yang kaya fitur dapat ditampilkan. Sementara itu, untuk menentukan rute, aplikasi memanfaatkan layanan *Directions API* yang memproses data koordinat asal dan tujuan untuk kemudian mengembalikan data *polyline* rute yang akan digambar di atas peta. Penyimpanan data lokasi secara lokal ditangani oleh *SQLite database* yang ringan dan efisien pada perangkat *mobile*, sedangkan proses *backup* dan *import* dilakukan dengan membaca dan menulis file berformat *JSON* atau *CSV* ke dalam penyimpanan perangkat, yang memastikan kompatibilitas dan kemudahan dalam proses transfer data.

Tabel 1: Fitur Aplikasi PwkTrack

No.	Fitur	Keterangan
1	Input data Lokasi	Pengguna dapat menandai lokasi
2	Tampilan rute	Menampilkan rute dari lokasi saat ini ke tujuan
3	<i>Backup</i> data	Menyimpan data lokasi ke penyimpanan eksternal

2. Studi Perbandingan: *Java* vs *Flutter*

Tabel 2: Perbandingan Pengembangan Aplikasi

Aspek	<i>Java (Android Native)</i>	<i>Flutter (Dart)</i>
Bahasa	<i>Java</i>	<i>Dart</i>
<i>UI Development</i>	<i>XML</i>	<i>Widget</i>
Integrasi <i>Maps</i>	<i>Google Maps API</i>	<i>google_maps_flutter</i>
Performa	Baik	Sangat Baik
Kemudahan Pengembangan	Sedang	Mudah

Kode *Flutter* untuk Menampilkan Peta

dart

```
import 'package:flutter/material.dart';
```

```
import 'package:google_maps_flutter/google_maps_flutter.dart';
```

```
void main() => runApp(MyApp());

class MyApp extends StatelessWidget {

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      home: Scaffold(

        appBar: AppBar(title: Text('Peta Flutter')),

        body: GoogleMap(

          initialCameraPosition: CameraPosition(

            target: LatLng(-0.7893, 113.9213),

            zoom: 5,

          ),

        ),

      ),

    );

  }

}
```

PEMBAHASAN

Pengembangan aplikasi PwkTrack dalam penelitian ini berhasil merepresentasikan implementasi praktis dari sistem *Mobile GIS* yang terintegrasi dengan layanan peta digital *modern*. Keberhasilan ini tidak terlepas dari pemanfaatan arsitektur teknologi yang tepat, yang menghubungkan secara sinergis antara komponen perangkat keras, perangkat lunak, dan layanan *cloud*.

Integrasi Teknologi sebagai Pilar Utama

Kemampuan aplikasi dalam menampilkan peta *digital* yang interaktif dan dinamis didukung sepenuhnya oleh *Google Maps API*. Integrasi ini merupakan pilihan yang strategis, karena *Google Maps* telah menjadi standar *de facto* untuk peta *digital* dengan basis data geospasial yang sangat lengkap dan *terupdate*.

API ini tidak hanya menyediakan tile peta (potongan gambar peta) tetapi juga data jalan, lalu lintas, dan *Points of Interest (POI)* yang kaya, yang menjadi fondasi bagi semua fitur navigasi aplikasi.

Sementara itu, akurasi dalam penentuan posisi pengguna dicapai melalui kombinasi teknologi *GPS* pada perangkat *smartphone* dan *Location Based Service (LBS)*. *GPS* memberikan koordinat geografis dengan akurasi tinggi di area terbuka, sedangkan *LBS*, yang memanfaatkan sinyal seluler dan *Wi-Fi*, berperan penting dalam melengkapi dan meningkatkan kecepatan penentuan posisi di dalam ruangan atau di area dengan cakupan satelit yang terbatas. Konstelasi satelit *GPS*, seperti yang diilustrasikan dalam landasan teori, bekerja untuk memberikan sinyal triangulasi yang kemudian diolah oleh perangkat untuk menghasilkan posisi yang akurat (Safaat, 2012).

Mekanisme Penentuan Rute dan Manajemen Data

Fitur penentuan rute kendaraan merupakan inti dari aplikasi ini. Mekanisme ini diaktifkan ketika pengguna memilih sebuah lokasi tujuan. Aplikasi kemudian mengambil koordinat pengguna (dari *GPS/LBS*) dan koordinat tujuan, kemudian mengirimkannya ke *Google Directions API* melalui permintaan jaringan (*network request*).

API tersebut akan memproses data ini dan mengembalikan respons dalam format yang terstruktur (biasanya *JSON*), yang berisi serangkaian koordinat (*polyline*) yang membentuk rute teroptimal, lengkap dengan data estimasi jarak dan waktu tempuh. Aplikasi kemudian menerjemahkan data *polyline* ini menjadi garis *visual* berwarna biru yang ditumpangkan di atas peta, memberikan panduan yang jelas dan mudah diikuti oleh pengguna (*Google Maps Platform*, 2023).

Di sisi manajemen data, aplikasi mengimplementasikan fungsi penyimpanan lokal yang efisien menggunakan *SQLite*. Database ini menyimpan data lokasi favorit pengguna, seperti nama lokasi, alamat, dan koordinat *latitude-longitude*. Pendekatan ini memungkinkan akses

data yang cepat tanpa ketergantungan pada koneksi *internet* yang konstan. Fitur *backup* dan *restore* yang ditawarkan merupakan nilai tambah yang signifikan untuk pengalaman pengguna.

Dengan mengeksport data *SQLite* ke dalam *file* berformat seperti *JSON* atau *CSV* ke penyimpanan eksternal, pengguna diberikan kendali penuh atas data mereka. Proses ini melindungi data dari kehilangan akibat kerusakan perangkat atau saat pengguna melakukan migrasi ke *smartphone* baru, sehingga meningkatkan loyalitas pengguna terhadap aplikasi (Flutter Documentation, 2023).

Studi Komparatif: *Android Native (Java)* vs. *Flutter (Dart)*
Penelitian ini juga memberikan wawasan berharga melalui studi perbandingan antara pengembangan aplikasi *native* menggunakan *Java* dan pendekatan *modern* menggunakan *Flutter* dengan bahasa *Dart*. Pengembangan *native* dengan *Java* di *Android Studio* (melalui *ADT Bundle*) memberikan keuntungan berupa akses yang mendalam dan langsung ke semua fitur *platform Android* serta *SDK Google Maps* yang telah matang. Hal ini terbukti dengan keberhasilan pembuatan PwkTrack yang stabil.

Namun, analisis terhadap materi pemrograman modern (Pertemuan 4 & 5) menunjukkan bahwa *Flutter* menawarkan paradigma pengembangan yang berbeda dengan keunggulan tertentu. *Flutter* mengadopsi pendekatan "*everything is a widget*" yang sangat *deklaratif*, sehingga mempermudah dan mempercepat pembuatan antarmuka pengguna (UI) yang konsisten dan kompleks.

Kode Dart untuk membuat tombol, *layout*, dan animasi cenderung lebih ringkas dan mudah dibaca dibandingkan dengan *Java/XML* tradisional. Keunggulan utama *Flutter* adalah kemampuan *single codebase*-nya, di mana kode yang sama dapat dikompilasi untuk *platform Android* dan *iOS*, yang secara drastis dapat mengurangi biaya dan waktu pengembangan. *Widget-widget layout* seperti *Column*, *Row*, *Stack*, dan *Container* yang dipelajari, memberikan fleksibilitas yang besar dalam menata tampilan tanpa perlu beralih konteks ke *file layout XML*. Meskipun pada tahap penelitian ini integrasi peta di *Flutter* memerlukan plugin tambahan (*google_maps_flutter*), ekosistem paket *Flutter* yang sangat aktif memastikan bahwa fungsionalitas inti seperti *GIS* dan peta dapat diimplementasikan dengan baik. Kendala dan Optimasi.

Dalam proses pengembangannya, tantangan teknis yang umum dihadapi adalah mengelola permintaan *API Key Google Maps* dan memastikan efisiensi baterai perangkat akibat penggunaan *GPS* yang intensif. Optimasi dilakukan dengan mengimplementasikan

pemanggilan *API* yang efisien serta manajemen siklus hidup *Activity Android* yang tepat untuk membebaskan sumber daya ketika tidak digunakan.

Dari sisi pengguna, aplikasi yang dibangun telah memenuhi kebutuhan dasarnya, namun untuk pengembangan lebih lanjut, fitur seperti pencarian lokasi *real-time* menggunakan *Google Places API* dan pemilihan moda transportasi yang lebih beragam dapat ditambahkan untuk meningkatkan utilitas aplikasi. Secara keseluruhan, penelitian ini membuktikan bahwa penggabungan teknologi *Mobile GIS*, *cloud service* seperti *Google Maps*, dan metodologi pengembangan aplikasi yang tepat—baik *native* maupun *cross-platform*—dapat menghasilkan sebuah solusi navigasi yang *powerful*, akurat, dan *user-friendly* untuk memecahkan masalah mobilitas sehari-hari.

Kesimpulan

1. Aplikasi *Mobile GIS* berbasis *Android* berhasil dikembangkan dengan memanfaatkan *Google Maps API*, *GPS*, dan *LBS*.
2. *Flutter* menawarkan kemudahan dalam pengembangan *UI* dan integrasi peta dibandingkan *Java*.
3. Aplikasi dapat berjalan pada perangkat *Android* dengan spesifikasi minimal *OS Gingerbread*.

DAFTAR PUSTAKA

- Flutter Documentation. (2023). <https://flutter.dev/docs>
- Google Maps Platform. (2023). <https://developers.google.com/maps>
- Riyanto. (2010). *Sistem Informasi Geografis Berbasis Mobile*. Yogyakarta.
- Safaat, N. (2012). *Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.