

**LAPORAN PRAKTIKUM MOBILE PROGRAMMING
MODUL 12**



Nama : Safri Firzan Sururi
NIM : 240605110084
Kelas : Mobile Programming B
Tanggal : 03 – 11 – 2025

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
GANJIL 2025/2026**

I. Tujuan

Tujuan dari praktikum ini adalah:

- Memahami konsep pengambilan lokasi perangkat menggunakan GPS pada framework Flutter.
- Menerapkan paket geolocator untuk mengambil koordinat dan geocoding untuk mengubah koordinat menjadi alamat (reverse geocoding).
- Memahami penanganan izin (permission handling) lokasi pada Android.
- Mengimplementasikan antarmuka yang interaktif dan responsif menggunakan State Management sederhana (setState) untuk menampilkan informasi lokasi secara real-time.

III. Langkah Kerja

1. Persiapan Dependensi dan Izin

Langkah pertama adalah menambahkan paket yang diperlukan ke dalam file pubspec.yaml, yaitu geolocator, geocoding, dan google_fonts. Setelah itu, dilakukan konfigurasi izin pada file android/app/src/main/AndroidManifest.xml dengan menambahkan ACCESS_FINE_LOCATION dan ACCESS_COARSE_LOCATION agar aplikasi diizinkan mengakses GPS perangkat.

2. Pembuatan Antarmuka (UI)

Antarmuka dibangun menggunakan widget Scaffold sebagai kerangka utama. Di dalamnya terdapat Card untuk menampilkan hasil lokasi dan ElevatedButton sebagai pemicu aksi. UI dirancang responsif terhadap status aplikasi:

- Menampilkan CircularProgressIndicator saat proses memuat (_isLoading bernilai true).
- Menampilkan pesan error jika terjadi kegagalan.
- Menampilkan teks "Tekan tombol..." jika belum ada data.
- Menampilkan Nama Kecamatan dan Kota jika data berhasil didapatkan.

3. Implementasi Logika Lokasi (_getLocation)

Fungsi utama dalam aplikasi ini adalah `_getLocation()`. Fungsi ini bekerja secara asynchronous untuk melakukan tugas-tugas berikut:

Reset Status: Mengubah `_isLoading` menjadi true dan menghapus data lama serta pesan error.

Cek Layanan & Izin: Memastikan GPS aktif (`isLocationServiceEnabled`) dan memeriksa izin lokasi (`checkPermission`). Jika izin ditolak, aplikasi akan memintanya melalui `requestPermission`.

Ambil Posisi: Menggunakan `Geolocator.getCurrentPosition()` untuk mendapatkan koordinat latitude dan longitude dengan presisi tinggi.

Reverse Geocoding: Mengonversi koordinat tersebut menjadi alamat (Kecamatan dan Kota) menggunakan fungsi `placemarkFromCoordinates()`.

4. Penjelasan Kode dan Variabel

Berikut adalah analisis penggunaan variabel dan fungsi dalam menampilkan informasi secara dinamis:

Variabel `_isLoading`: Bertipe boolean. Variabel ini berfungsi sebagai "saklar" indikator. Saat true, UI akan merender loading spinner, mencegah pengguna menekan tombol berkali-kali, dan memberi tanda bahwa aplikasi sedang bekerja.

Variabel `_kecamatan` dan `_kota`: Bertipe String nullable (String?). Variabel ini menyimpan hasil sukses dari reverse geocoding. Data inilah yang ditampilkan di dalam widget Text pada Card.

Variabel `_errorMessage`: Digunakan untuk menampung pesan kesalahan (exception), misalnya jika GPS mati atau izin ditolak. Jika variabel ini tidak null, teks error berwarna merah akan muncul di layar.

Fungsi `setState()`: Ini adalah inti dari reaktivitas tampilan. Flutter tidak akan tahu bahwa data (`_kecamatan`, dll) telah berubah tanpa perintah ini. `setState()` memerintahkan Flutter untuk membangun ulang (rebuild) widget, sehingga tampilan layar diperbarui secara otomatis sesuai dengan data terbaru (misalnya dari loading menjadi teks alamat).

IV. Screenshot Hasil

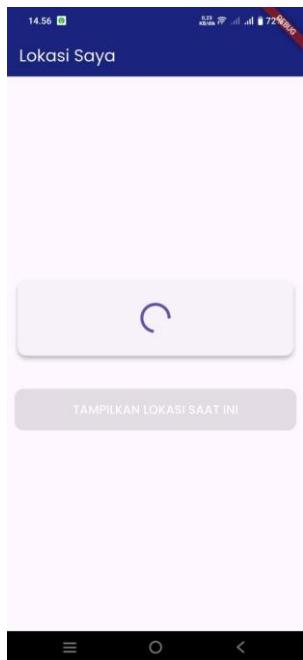
Menyertakan tangkapan layar aplikasi untuk berbagai kondisi, misalnya: tampilan awal sebelum lokasi ditampilkan, saat pemrosesan lokasi berlangsung, setelah lokasi berhasil ditampilkan, dan saat muncul pesan error. Setiap gambar diberi keterangan singkat untuk memudahkan pemahaman.

Kondisi Tampilan Awal :



Tampilan saat aplikasi baru dibuka. Belum ada data lokasi, pengguna diminta menekan tombol.

Kondisi Proses Loading :



Tampilan saat tombol ditekan. `_isLoading` aktif, indikator berputar muncul.

Kondisi Lokasi Berhasil :



Tampilan sukses. Menampilkan data Kecamatan dan Kota sesuai posisi GPS.

Kondisi Error :



Tampilan jika GPS dimatikan atau izin ditolak. Muncul pesan error berwarna merah.

IV. Kesimpulan

Berdasarkan praktikum yang telah dilakukan, dapat disimpulkan bahwa:

- Aplikasi Flutter berhasil mengakses fitur native Android (GPS) menggunakan plugin geolocator untuk mendapatkan koordinat dan geocoding untuk mendapatkan detail alamat.
- Fungsi `_getLocation()` berperan sentral dalam mengorkestrasi alur data: mulai dari validasi izin, pengambilan data, hingga penanganan error (try-catch block).
- Penggunaan `setState()` sangat krusial dalam aplikasi stateful. Tanpa method ini, perubahan nilai pada variabel `_kecamatan` atau `_kota` tidak akan tercermin pada antarmuka pengguna.
- Penerapan logika UI kondisional (menggunakan operator ternary `? :`) memungkinkan satu layar menangani berbagai status (idle, loading, success, error) dengan rapi dan informatif bagi pengguna.