Mobile Progreming Lanjut

# “Aplikasi Prakiraan Cuaca BMKG – Sago Salido, Pesisir Selatan

“



**Dosen pengampu**

Ade Kurniawan, S.Pd., M.Pd.T

**Di susun oleh:**

TAUFIQUL ISRAT 22346026

**FAKULTAS TEKNIK DEPARTEMEN ELEKTRONIKA**

**PRODI INFORMATIKA UNIVERSITAS NEGERI**

**PADANG**

# Aplikasi Prakiraan Cuaca BMKG – Sago Salido, Pesisir Selatan

## 1. Latar Belakang

Cuaca merupakan informasi penting dalam kehidupan sehari-hari, terutama bagi masyarakat pesisir yang bergantung pada kondisi alam. Oleh karena itu, dibutuhkan aplikasi yang dapat menyajikan informasi prakiraan cuaca secara real-time dan mudah diakses. Aplikasi ini memanfaatkan API dari BMKG dan menampilkan data cuaca untuk wilayah Sago Salido, Pesisir Selatan.

## 2. Tujuan

- Membuat aplikasi mobile dengan Flutter untuk menampilkan prakiraan cuaca.

- Mengambil data cuaca secara real-time dari API BMKG.

- Menyajikan antarmuka yang bersih, responsif, dan mudah dipahami.

## 3. Teknologi yang Digunakan

- Flutter

- Dart

- HTTP Package

- API BMKG

## 4. Struktur Proyek

lib/

├── developer\_page.dart # Halaman profil developer

├── main.dart # Entry point dan logic utama

├── style.dart # Tema dan gaya aplikasi

└── weather\_card.dart # Komponen kartu cuaca modular

## 5. Fitur Aplikasi

- Tampilan Cuaca: Grid dinamis berisi informasi tanggal, ikon cuaca, deskripsi, suhu, dan jam prakiraan.

- Responsive UI: Menyesuaikan jumlah kolom dan ukuran elemen berdasarkan lebar layar.

- Animasi: Ikon cuaca memiliki efek animasi naik-turun untuk menarik perhatian pengguna.

- Profil Developer: Halaman khusus menampilkan informasi pengembang.

- Error Handling: Menampilkan pesan kesalahan saat gagal mengambil data dari API.

## 6. Penjelasan File Utama

* - main.dart

Menangani permintaan HTTP ke API BMKG, parsing data cuaca, dan menampilkannya dalam grid view. Juga mengatur navigasi ke halaman developer.

import 'dart:convert';

import 'package:flutter/material.dart';

import 'package:http/http.dart' as http;

import 'developer\_page.dart';

import 'style.dart';

void main() {

  runApp(const BMKGWeatherApp());

}

class BMKGWeatherApp extends StatelessWidget {

  const BMKGWeatherApp({super.key});

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      title: 'Prakiraan Cuaca BMKG',

      debugShowCheckedModeBanner: false,

      theme: appTheme,

      routes: {'/developer': (context) => const DeveloperPage()},

      home: const WeatherHomePage(),

    );

  }

}

class WeatherHomePage extends StatefulWidget {

  const WeatherHomePage({super.key});

  @override

  State<WeatherHomePage> createState() => \_WeatherHomePageState();

}

class \_WeatherHomePageState extends State<WeatherHomePage>

    with SingleTickerProviderStateMixin {

  List<dynamic> weatherList = [];

  bool loading = true;

  String error = '';

  late final AnimationController \_animationController;

  late final Animation<double> \_animation;

  @override

  void initState() {

    super.initState();

    fetchWeather();

    \_animationController = AnimationController(

      vsync: this,

      duration: const Duration(seconds: 2),

    )..repeat(reverse: true);

    \_animation = Tween<double>(begin: 0, end: 8).animate(

      CurvedAnimation(parent: \_animationController, curve: Curves.easeInOut),

    );

  }

  @override

  void dispose() {

    \_animationController.dispose();

    super.dispose();

  }

  Future<void> fetchWeather() async {

    final url = Uri.parse(

      'https://api.bmkg.go.id/publik/prakiraan-cuaca?adm4=13.01.05.2003',

    );

    try {

      final response = await http.get(url);

      if (response.statusCode == 200) {

        final jsonData = json.decode(response.body);

        final List<dynamic> cuaca = jsonData['data'][0]['cuaca'];

        final List<dynamic> cuacaFlat =

            cuaca.expand((e) => e is List ? e : [e]).toList();

        setState(() {

          weatherList = cuacaFlat;

          loading = false;

          error = '';

        });

      } else {

        setState(() {

          loading = false;

          error = 'Gagal mengambil data: ${response.statusCode}';

        });

      }

    } catch (e) {

      setState(() {

        loading = false;

        error = 'Error: $e';

      });

    }

  }

  String weatherEmoji(String desc) {

    final d = desc.toLowerCase();

    if (d.contains('cerah')) return '☀️';

    if (d.contains('berawan')) return '☁️';

    if (d.contains('hujan petir')) return '⛈️';

    if (d.contains('hujan ringan')) return '🌦️';

    if (d.contains('hujan sedang')) return '🌧️';

    if (d.contains('hujan lebat')) return '🌧️';

    if (d.contains('kabut')) return '🌫️';

    if (d.contains('bercak kabut')) return '🌫️';

    if (d.contains('berkabut')) return '🌫️';

    if (d.contains('salju')) return '❄️';

    return '';

  }

  @override

  Widget build(BuildContext context) {

    final screenWidth = MediaQuery.of(context).size.width;

    // Sesuaikan jumlah kolom grid berdasarkan lebar layar

    int crossAxisCount;

    if (screenWidth >= 600) {

      crossAxisCount = 7;

    } else if (screenWidth >= 400) {

      crossAxisCount = 5;

    } else {

      crossAxisCount = 2; // untuk layar kecil, 2 kolom agar tidak terlalu kecil

    }

    // Sesuaikan ukuran emoji berdasarkan layar

    double emojiSize;

    if (screenWidth >= 600) {

      emojiSize = 48;

    } else if (screenWidth >= 400) {

      emojiSize = 36;

    } else {

      emojiSize = 32; // sedikit lebih besar supaya tetap terbaca

    }

    // Padding responsive untuk layar kecil

    EdgeInsetsGeometry padding =

        screenWidth < 400 ? const EdgeInsets.all(8) : const EdgeInsets.all(12);

    return Scaffold(

      appBar: AppBar(

        title: const Text(

          'Prakiraan Cuaca BMKG Sumbar – Pesisir Selatan, Sago Salido',

        ),

        actions: [

          IconButton(

            icon: const Icon(Icons.person),

            tooltip: 'Developer',

            onPressed: () {

              Navigator.pushNamed(context, '/developer');

            },

          ),

        ],

      ),

      body:

          loading

              ? const Center(child: CircularProgressIndicator())

              : error.isNotEmpty

              ? Center(child: Text(error))

              : Padding(

                padding: padding,

                child: GridView.builder(

                  itemCount: weatherList.length,

                  gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(

                    crossAxisCount: crossAxisCount,

                    childAspectRatio: 0.65, // agar kartu agak lebih tinggi

                    crossAxisSpacing: 10,

                    mainAxisSpacing: 10,

                  ),

                  itemBuilder: (context, index) {

                    final item = weatherList[index];

                    final rawTime = item['local\_datetime'] ?? '';

                    final parsedTime =

                        DateTime.tryParse(rawTime) ?? DateTime.now();

                    final formattedDate =

                        '${parsedTime.day}/${parsedTime.month}';

                    final formattedHour =

                        '${parsedTime.hour.toString().padLeft(2, '0')}:00';

                    final temp = item['t']?.toString() ?? '-';

                    final desc = item['weather\_desc'] ?? '-';

                    return Card(

                      margin: const EdgeInsets.all(4),

                      child: Padding(

                        padding: const EdgeInsets.all(10),

                        child: Column(

                          mainAxisAlignment: MainAxisAlignment.center,

                          children: [

                            Text(

                              formattedDate,

                              style: Theme.of(context).textTheme.labelMedium

                                  ?.copyWith(fontWeight: FontWeight.bold),

                            ),

                            const SizedBox(height: 6),

                            AnimatedBuilder(

                              animation: \_animation,

                              builder: (context, child) {

                                return Transform.translate(

                                  offset: Offset(0, -\_animation.value),

                                  child: child,

                                );

                              },

                              child: Text(

                                weatherEmoji(desc),

                                style: TextStyle(fontSize: emojiSize),

                              ),

                            ),

                            const SizedBox(height: 6),

                            Text(

                              desc,

                              textAlign: TextAlign.center,

                              style: Theme.of(context).textTheme.bodyMedium,

                            ),

                            const SizedBox(height: 6),

                            Text(

                              '$temp °C',

                              style: Theme.of(context).textTheme.headlineSmall,

                            ),

                            const SizedBox(height: 6),

                            Text(

                              formattedHour,

                              style: Theme.of(context).textTheme.labelMedium,

                            ),

                          ],

                        ),

                      ),

                    );

                  },

                ),

              ),

    );

  }

}

* - weather\_card.dart

Komponen stateless yang bertanggung jawab untuk menampilkan satu elemen prakiraan cuaca dalam bentuk kartu.

import 'package:flutter/material.dart';

class WeatherCard extends StatelessWidget {

  final String formattedDate;

  final String emoji;

  final String description;

  final String temperature;

  final String formattedHour;

  final Animation<double> animation;

  const WeatherCard({

    super.key,

    required this.formattedDate,

    required this.emoji,

    required this.description,

    required this.temperature,

    required this.formattedHour,

    required this.animation,

  });

  @override

  Widget build(BuildContext context) {

    return Card(

      child: Padding(

        padding: const EdgeInsets.all(6.0),

        child: Column(

          mainAxisAlignment: MainAxisAlignment.center,

          children: [

            Text(

              formattedDate,

              style: Theme.of(

                context,

              ).textTheme.labelMedium?.copyWith(fontWeight: FontWeight.bold),

            ),

            const SizedBox(height: 4),

            AnimatedBuilder(

              animation: animation,

              builder: (context, child) {

                return Transform.translate(

                  offset: Offset(0, -animation.value),

                  child: child,

                );

              },

              child: Text(emoji, style: const TextStyle(fontSize: 36)),

            ),

            const SizedBox(height: 4),

            Text(

              description,

              textAlign: TextAlign.center,

              style: Theme.of(context).textTheme.bodyMedium,

            ),

            const SizedBox(height: 4),

            Text(

              '$temperature °C',

              style: Theme.of(context).textTheme.headlineSmall,

            ),

            const SizedBox(height: 4),

            Text(formattedHour, style: Theme.of(context).textTheme.labelMedium),

          ],

        ),

      ),

    );

  }

}

* - style.dart

Berisi definisi tema dan gaya untuk konsistensi tampilan UI aplikasi.

import 'package:flutter/material.dart';

final ThemeData appTheme = ThemeData(

  primaryColor: const Color(0xFF2A3A54), // biru gelap modern

  colorScheme: ColorScheme.fromSwatch().copyWith(

    secondary: const Color(0xFF91A6FF), // warna aksen pastel biru

  ),

  brightness: Brightness.light,

  scaffoldBackgroundColor: const Color(

    0xFFF5F7FA,

  ), // background putih kebiruan lembut

  appBarTheme: const AppBarTheme(

    backgroundColor: Color(0xFF2A3A54),

    centerTitle: true,

    elevation: 6,

    titleTextStyle: TextStyle(

      fontSize: 22,

      fontWeight: FontWeight.bold,

      color: Colors.white,

      letterSpacing: 1.2,

    ),

    iconTheme: IconThemeData(color: Colors.white),

  ),

  cardTheme: CardTheme(

    color: Colors.white,

    shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(24)),

    elevation: 12,

    margin: const EdgeInsets.symmetric(horizontal: 12, vertical: 8),

    shadowColor: Colors.black.withOpacity(0.15),

  ),

  elevatedButtonTheme: ElevatedButtonThemeData(

    style: ElevatedButton.styleFrom(

      backgroundColor: const Color(0xFF91A6FF),

      foregroundColor: Colors.white,

      shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(20)),

      textStyle: const TextStyle(fontSize: 16, fontWeight: FontWeight.w600),

      elevation: 8,

      shadowColor: Colors.blueAccent.withOpacity(0.3),

    ),

  ),

  inputDecorationTheme: InputDecorationTheme(

    border: OutlineInputBorder(

      borderRadius: BorderRadius.circular(20),

      borderSide: BorderSide(color: Colors.grey.shade300),

    ),

    focusedBorder: OutlineInputBorder(

      borderRadius: BorderRadius.circular(20),

      borderSide: const BorderSide(color: Color(0xFF91A6FF), width: 2),

    ),

    labelStyle: const TextStyle(

      color: Color(0xFF475569),

      fontWeight: FontWeight.w600,

      letterSpacing: 0.5,

    ),

  ),

  textTheme: const TextTheme(

    labelMedium: TextStyle(

      fontSize: 14,

      color: Color(0xFF334155),

      fontWeight: FontWeight.w700,

      letterSpacing: 0.8,

    ),

    bodyMedium: TextStyle(

      fontSize: 13,

      color: Color(0xFF64748B),

      height: 1.4,

      fontWeight: FontWeight.w500,

    ),

    headlineSmall: TextStyle(

      fontSize: 20,

      fontWeight: FontWeight.bold,

      color: Color(0xFF1E293B),

      letterSpacing: 1,

    ),

  ),

);

* - developer\_page.dart

Menampilkan halaman profil developer termasuk foto, nama, dan peran.

// lib/developer\_page.dart

import 'package:flutter/material.dart';

class DeveloperPage extends StatelessWidget {

  const DeveloperPage({super.key});

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: const Text('Developer'),

        backgroundColor: Theme.of(context).primaryColor,

        centerTitle: true,

      ),

      body: Center(

        child: Column(

          mainAxisAlignment: MainAxisAlignment.center,

          children: [

            ClipOval(

              child: Image.asset(

                'assets/images/developer.jpg',

                width: 140,

                height: 140,

                fit: BoxFit.cover,

              ),

            ),

            const SizedBox(height: 20),

            const Text(

              '22346026 TAUFIQUL ISRAT',

              style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),

            ),

            const SizedBox(height: 8),

            const Text(

              'Mobile Developer & Weather Enthusiast',

              style: TextStyle(fontSize: 16, color: Colors.black54),

            ),

          ],

        ),

      ),

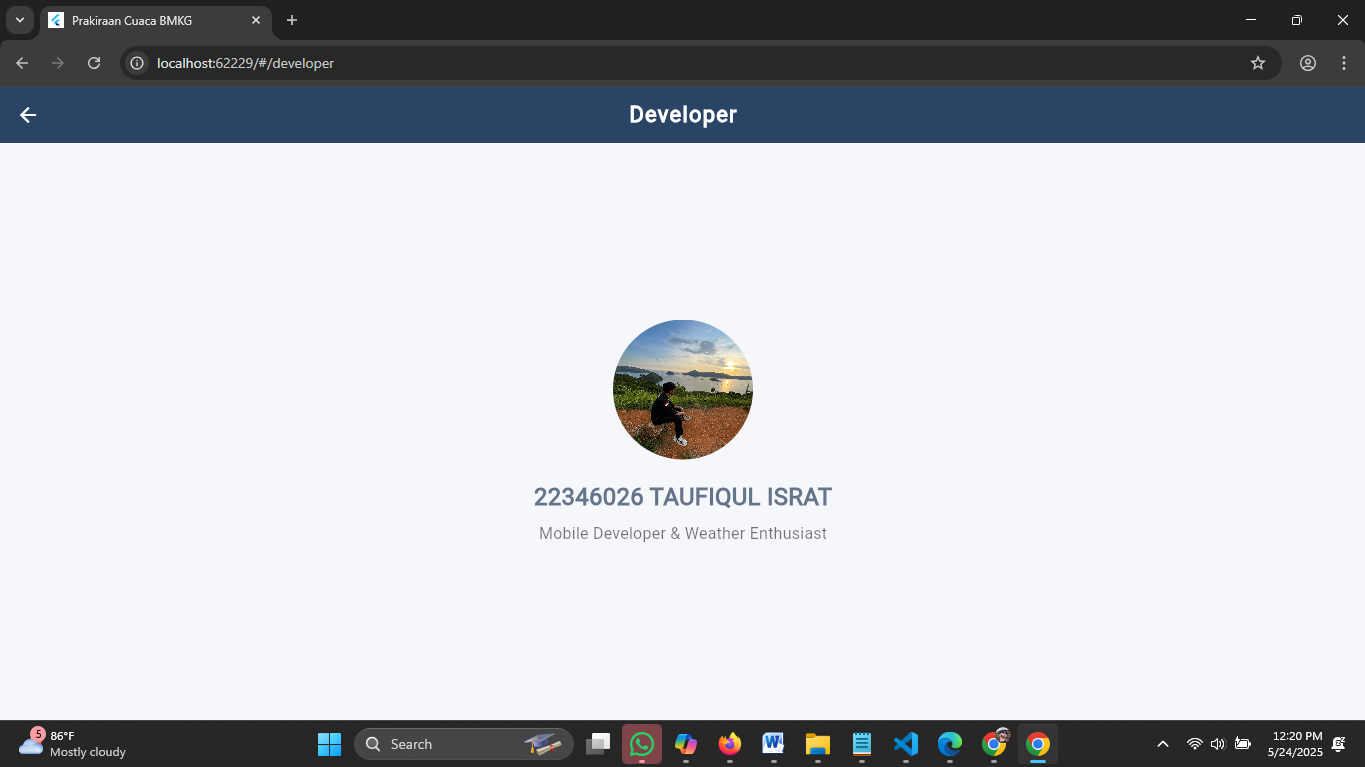
    );

  }

} // End of developer\_page.dart

Output





Bentuk mobile

