

LAPORAN TEORI MOBILE PROGRAMMING
MODUL 6



Nama : Erai Bagusalm
NIM : 240605110088
Kelas : Mobile Programming B
Tanggal : 22 September 2025

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

i. **Tujuan**

1. Memahami penggunaan widget GridView untuk menampilkan data dalam bentuk kotak-kotak (grid).
2. Memahami penggunaan widget ListView untuk menampilkan data dalam bentuk daftar yang dapat digulir (scrollable).
3. Mampu membuat tampilan aplikasi yang interaktif menggunakan GridTile, ListTile, dan properti pendukung seperti padding, warna, dan ikon.
4. Mampu menggunakan GridView.builder untuk menampilkan data dalam format grid secara dinamis dan efisien.
5. Mampu menggunakan ListView.builder untuk menampilkan data dalam format list (daftar) yang panjang dengan pengelolaan memori lebih baik.

ii. **Langkah Kerja**

1. Gunakan proyek Flutter **demo_listview_builder**.
2. Tambahkan dependensi **http: ^1.5** pada pubspec.yaml dan jalankan flutter pub get.
3. Ubah fungsi **_ambilData()** untuk mengambil data dari **API FreeToGame**.
4. Simpan hanya **20 data pertama** dari hasil API ke dalam list.
5. Buat fungsi **_tombolBaca()** untuk menampilkan tombol “Baca Info”.
6. Buat fungsi **_listItem()** untuk menampilkan gambar, judul, genre, tanggal rilis, dan tombol.
7. Gunakan **ListView.builder** untuk menampilkan daftar game dengan **_listItem**.
8. Jalankan aplikasi dan pastikan daftar game tampil dengan rapi di layar.

iii. Screenshot Hasil

a. Kode Program

```
1 import 'package:flutter/material.dart';
2 import 'dart:convert';
3 import 'package:http/http.dart' as http;
4
5 Run | Debug | Profile
6 void main() => runApp(const MyApp());
7
8 class MyApp extends StatelessWidget {
9   const MyApp({super.key});
10
11   @override
12   Widget build(BuildContext context) {
13     return MaterialApp(
14       title: 'Demo GridView.builder',
15       theme: ThemeData(
16         colorScheme: ColorScheme.fromSeed(seedColor: Colors.amber),
17         useMaterial3: true,
18       ), // ThemeData
19       home: const MyHomePage(title: 'Demo GridView.builder'),
20     ); // MaterialApp
21   }
22
23   class MyHomePage extends StatefulWidget {
24     const MyHomePage({super.key, required this.title});
25
26     final String title;
27
28     @override
29     State<MyHomePage> createState() => _MyHomePageState();
30   }
31
32   class _MyHomePageState extends State<MyHomePage> {
33     // List untuk menyimpan data berita
34     List dataGame = [];
35
36     @override
37     void initState() {
38       super.initState();
39       _ambilData();
40     }
41   }
42 }
```

```
42 // Method untuk request data ke server dengan API FreeToGame
43 Future _ambilData() async {
44   try {
45     final response = await http.get(
46       Uri.parse('https://www.freetogame.com/api/games'),
47     );
48
49     if (response.statusCode == 200) {
50       final data = jsonDecode(response.body);
51
52       setState(() {
53         // Simpan hanya 10 data pertama dari API ke List
54         dataGame = data.take(20).toList();
55       });
56     } else {
57       throw Exception('Gagal load data dari FreeToGame API');
58     }
59   } catch (e) {
60     print('Error: $e'); // Don't invoke 'print' in production code, try using a
61   }
62 }
63
64 @override
65 Widget build(BuildContext context) {
66   return Scaffold(
67     appBar: AppBar(backgroundColor: Colors.amber, title: Text(widget.title)),
68     body: ListView.builder(
69       itemCount: dataGame.length,
70       itemBuilder: (context, index) {
71         return _listItem(
72           dataGame[index]['thumbnail'] ?? 'https://via.placeholder.com/150',
73           dataGame[index]['title'] ?? 'Tidak ada judul',
74           dataGame[index]['genre'] ?? 'Tidak ada genre',
75           dataGame[index]['release_date'] ?? 'Tidak ada tanggal',
76         );
77       }, // ListView.builder
78     ); // Scaffold
79   }
80 }
81 }
```

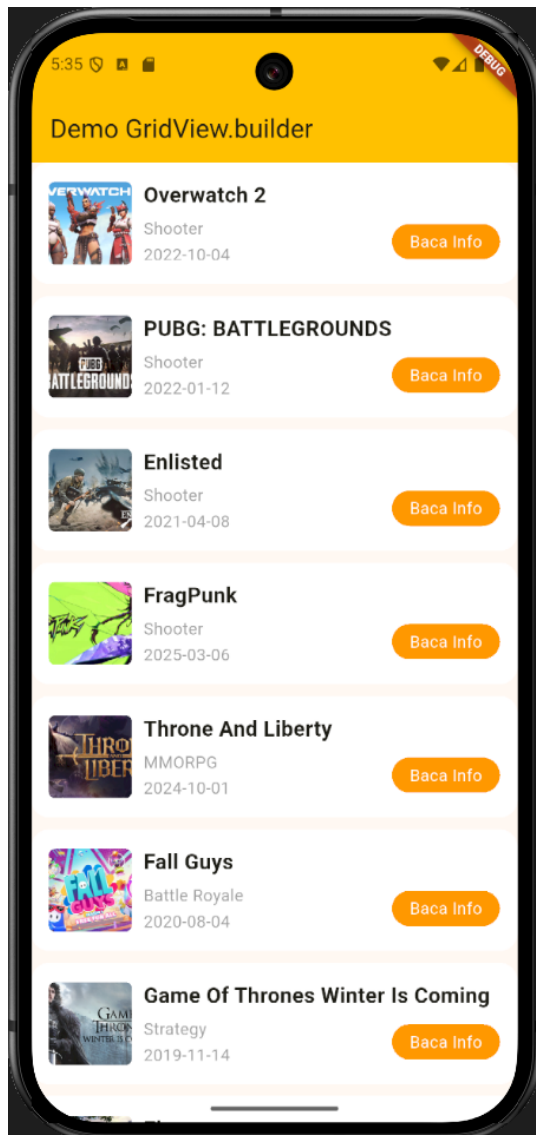
```
83 Container _tomolBaca() {
84   return Container(
85     padding: const EdgeInsets.symmetric(vertical: 5, horizontal: 15),
86     decoration: BoxDecoration(
87       color: Colors.orange,
88       borderRadius: BorderRadius.circular(15),
89     ), // BoxDecoration
90     child: const Text('Baca Info', style: TextStyle(color: Colors.white)),
91   ); // Container
92 }
93
94 Container _listItem(String url, String judul, String genre, String rilis) {
95   return Container(
96     padding: const EdgeInsets.all(15),
97     margin: const EdgeInsets.only(bottom: 10),
98     decoration: BoxDecoration(
99       color: Colors.white,
100       borderRadius: BorderRadius.circular(15),
101     ), // BoxDecoration
102     child: Row(
103       children: [
104         ClipRect(
105           borderRadius: BorderRadius.circular(5),
106           child: Image.network(url, width: 70, height: 70, fit: BoxFit.cover),
107         ), // ClipRect
108         const SizedBox(width: 10),
109         Expanded(
110           child: Column(
111             crossAxisAlignment: CrossAxisAlignment.start,
112             children: [
113               Text(
114                 judul,
115                 style: const TextStyle(
116                   fontSize: 18,
117                   fontWeight: FontWeight.bold,
118                 ), // TextStyle
119             ), // Text
120             const SizedBox(height: 5),
121             Row(
122               mainAxisAlignment: MainAxisAlignment.spaceBetween,
123               children: [
124                 Column(
125                   crossAxisAlignment: CrossAxisAlignment.start,
126                   children: [
127                     Text(genre, style: const TextStyle(color: Colors.grey)),
128                     const SizedBox(height: 2),
129                     Text(rilis, style: const TextStyle(color: Colors.grey)),
130                   ], // Column
131                 ), // Column
132                 _tomolBaca(),
133               ], // Row
134             ), // Row
135           ), // Expanded
136         ), // Column
137       ], // Row
138     ); // Container
139   ); // Container
140 }
141 }
```

b. Penjelasan Kode Program

Program ini dibuat menggunakan Flutter untuk menampilkan daftar game yang diambil dari API publik FreeToGame. Proses dimulai dengan menambahkan dependensi HTTP agar aplikasi dapat melakukan permintaan (request) data secara online. Pada fungsi `_ambilData()`, aplikasi mengirimkan request ke API dan menerima data berupa format JSON, lalu hanya 20 data pertama yang disimpan ke dalam list untuk ditampilkan.

Selanjutnya, dibuat fungsi `_tombolBaca()` yang berfungsi menampilkan tombol “Baca Info” berwarna oranye dengan teks putih agar mudah dikenali pengguna. Kemudian fungsi `_listItem()` dibuat untuk menampilkan setiap item game secara rapi menggunakan kombinasi widget Container, Row, dan Column. Setiap item menampilkan gambar thumbnail game, judul, genre, tanggal rilis, dan tombol aksi. Akhirnya, data game ditampilkan secara dinamis menggunakan `ListView.builder`, yang akan memanggil fungsi `_listItem()` untuk setiap elemen dalam daftar. Penggunaan `ListView.builder` membuat tampilan lebih efisien karena hanya item yang terlihat di layar yang akan dirender, sehingga performa aplikasi tetap ringan meskipun data banyak.

Output :



iv. Kesimpulan

Dari percobaan ini dapat disimpulkan bahwa widget `ListView.builder` sangat efektif untuk menampilkan data dalam jumlah banyak secara dinamis dan efisien. Dengan memanfaatkan **API FreeToGame**, aplikasi mampu menampilkan daftar game lengkap dengan gambar, judul, genre, serta tanggal rilis secara otomatis tanpa input manual. Penggunaan fungsi `_listItem()` membantu menjaga konsistensi desain setiap elemen daftar, sementara `_tombolBaca()` menambah elemen interaktif pada

tampilan. Secara keseluruhan, tugas ini memperlihatkan bagaimana Flutter dapat digunakan untuk menghubungkan aplikasi dengan sumber data online dan menampilkan informasi secara menarik, rapi, serta responsif.