

# **PRATIK APLIKASI MOBILE**

“Perbaikan Laporan Kuis Mobile”



**Disusun oleh :**

Diah Munica Nawang  
V3922015 / TI D

**Dosen :**

Trisna Ari Roshinta, S.S.T., M.T

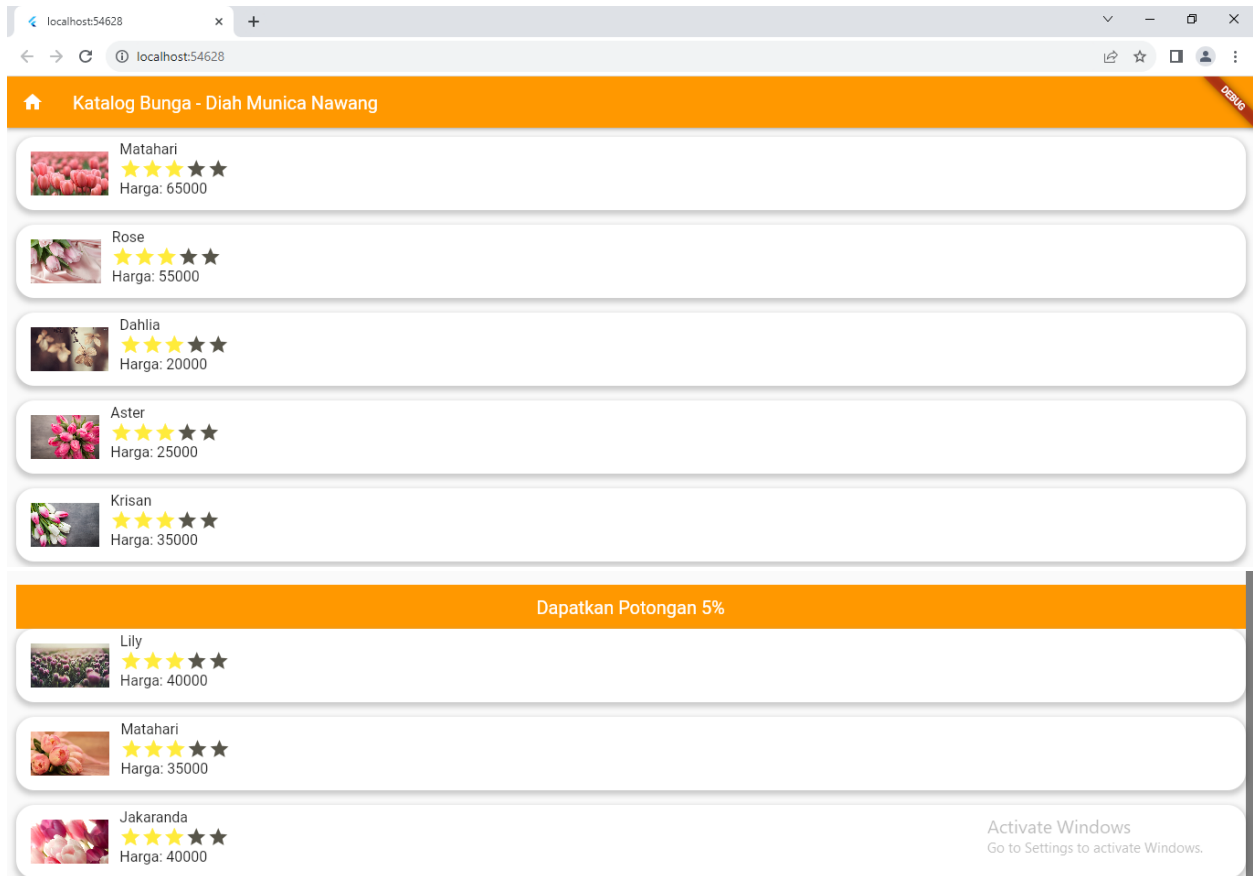
**PS D-III TEKNIK INFORMATIKA  
SEKOLAH VOKASI  
UNIVERSITAS SEBELAS MARET  
2023**

## Soal :

- Membuat laporan berisi penjelasan lengkap per baris source code nya dan screenshot hasilnya

## Jawab :

### Hasil Katalog Bunga :



## Hasil Script Perbaikan Serta Penjelasan Setiap Line :



```
1 import 'package:flutter/material.dart';  
2 //Mengimport library Flutter yang diperlukan  
3
```



```
1 void main() {  
2 //Fungsi main() yang menjalankan aplikasi  
3   runApp(MyApp());  
4 }  
5 //runApp() dengan widget MyApp() sebagai root aplikasi.
```

```

1 class MyApp extends StatelessWidget {
2   //Kelas MyApp tipe StatelessWidget.berarti MyApp adalah widget yang tidak dapat diubah setelah dibuat.
3   @override
4   //override menandakan bahwa metode di kelas anak menggantikan metode dengan nama yang sama di kelas induk atau antarmuka.
5   Widget build(BuildContext context) {
6     // untuk membangun tampilan dan mengembalikan widget yang akan ditampilkan pada layar.
7     return MaterialApp(
8       //digunakan sebagai akar (root) aplikasi Flutter untuk menyediakan fitur dan tampilan berdasarkan Material Design.
9       home: Scaffold(
10        //sebagai kerangka dasar yang berisi elemen-elemen seperti AppBar dan body.
11        appBar: AppBar(
12          //mengkonfigurasi bilah atas (appBar) dalam widget
13          backgroundColor: Colors.orange,
14          //mengatur warna latar belakang dari AppBar menjadi oranye.
15          title: Text('Katalog Bunga - Diah Munica Nawang'),
16          //mengatur teks judul yang akan ditampilkan di AppBar menjadi "Katalog Bunga - Diah Munica Nawang".
17          leading: Icon(Icons.home),
18          //menambahkan ikon rumah (home) sebagai elemen "leading" di dalam AppBar.
19        ),
20        body: FlowerList(),
21        //daftar bunga yang telah didefinisikan akan ditampilkan sebagai konten utama dalam aplikasi.
22      ),
23    );
24  }
25 }

```

```

1 class FlowerList extends StatelessWidget {
2   //digunakan untuk membuat daftar bunga dalam aplikasi Flutter.
3   final List<Map> bunga = [
4     //sebagai daftar (List) yang berisi map (key-value pairs) yang merepresentasikan data bunga dalam aplikasi.
5     {"nama": "Matahari", "harga": "65000", "gambar": "1.jpg"},
6     {"nama": "Rose", "harga": "55000", "gambar": "2.jpg"},
7     {"nama": "Dahlia", "harga": "20000", "gambar": "3.jpg"},
8     {"nama": "Aster", "harga": "25000", "gambar": "4.jpg"},
9     {"nama": "Krisan", "harga": "35000", "gambar": "5.jpg"},
10    {"nama": "Lily", "harga": "40000", "gambar": "6.jpg"},
11    {"nama": "Matahari", "harga": "35000", "gambar": "7.jpg"},
12    {"nama": "Jakaranda", "harga": "40000", "gambar": "8.jpg"},
13  ];
14  //Baris-baris ini mendefinisikan beberapa data bunga dengan properti nama, harga, dan nama
15  //berkas gambar yang akan digunakan dalam aplikasi.

```



```
1 return Container(  
2 //gunakan untuk mengembalikan widget Container yang akan mengelilingi  
3 //dan memuat konten dari setiap item dalam daftar  
4   height: 80, //mengatur tinggi (height) dari widget  
5   decoration: BoxDecoration( //digunakan untuk mengatur dekorasi (style)  
6     borderRadius: BorderRadius.all(Radius.circular(20)), //mengatur sudut (radius) dari border (pinggiran)  
7     color: Colors.white, //mengatur warna  
8     boxShadow: [ //digunakan untuk menambahkan bayangan (shadow)  
9       BoxShadow( //mengatur warna, sebaran, kekaburan, dan offset bayangan.  
10        color: Colors.grey.withOpacity(0.5), // Warna shadow  
11        spreadRadius: 2, // Seberapa besar shadow  
12        blurRadius: 4, // Seberapa kabur shadow  
13        offset: Offset(0, 3), // Posisi shadow  
14      ),  
15    ],  
16  ),
```



```
1 return Container(  
2 //gunakan untuk mengembalikan widget Container yang akan mengelilingi  
3 //dan memuat konten dari setiap item dalam daftar  
4   height: 80, //mengatur tinggi (height) dari widget  
5   decoration: BoxDecoration( //digunakan untuk mengatur dekorasi (style)  
6     borderRadius: BorderRadius.all(Radius.circular(20)), //mengatur sudut (radius) dari border (pinggiran)  
7     color: Colors.white, //mengatur warna  
8     boxShadow: [ //digunakan untuk menambahkan bayangan (shadow)  
9       BoxShadow( //mengatur warna, sebaran, kekaburan, dan offset bayangan.  
10        color: Colors.grey.withOpacity(0.5), // Warna shadow  
11        spreadRadius: 2, // Seberapa besar shadow  
12        blurRadius: 4, // Seberapa kabur shadow  
13        offset: Offset(0, 3), // Posisi shadow  
14      ),  
15    ],  
16  ),
```



```
1 child: ListTile(  
2   //untuk menampilkan konten item dalam daftar dengan tata letak dan elemen yang telah ditentukan.  
3   leading: Image.asset(gambar),  
4   //mengatur gambar yang akan ditampilkan  
5   title: Column(  
6     //digunakan untuk menambahkan widget Column sebagai judul (title) dalam ListTile  
7     crossAxisAlignment: CrossAxisAlignment.start,  
8     //mengatur tata letak (layout) dari widget dalam  
9   ),
```

```

1  children: [
2      //digunakan untuk menambahkan sejumlah widget sebagai anak-anak (children)
3      Text(nama), //menambahkan teks yang akan ditampilkan
4      Row( //digunakan untuk membuat baris horizontal
5          mainAxisAlignment: MainAxisAlignment.min, //mengatur ukuran utama (main axis) dari Row agar sesempit mungkin,
6          children: [ //digunakan untuk menambahkan sejumlah widget sebagai anak-anak (children) dalam Row,
7              Icon(Icons.star, color: Colors.yellow),
8              Icon(Icons.star, color: Colors.yellow),
9              Icon(Icons.star, color: Colors.yellow),
10             Icon(Icons.star, color: Color.fromARGB(255, 87, 85, 74)),
11             Icon(Icons.star, color: Color.fromARGB(255, 87, 85, 74)),
12             //Bagian ini menambahkan beberapa ikon bintang ke dalam 'Row' sebagai anak-anak (children) dalam baris horizontal.
13             //Ikon-ikon ini digunakan untuk menampilkan peringkat item dalam bentuk bintang, dengan beberapa ikon bintang diwarnai kuning (yellow)
14             //dan beberapa ikon bintang diwarnai sesuai dengan kode warna RGB yang ditentukan.
15         ],
16     ),
17     Text('Harga: $harga'),
18     //menampilkan teks "Harga:" diikuti oleh nilai harga yang diambil dari variabel harga.
19 ],
20 ),
21 ),
22 );
23 },

```

```

1  separatorBuilder: (context, index) {
2      // digunakan untuk membangun pemisah antara item-item dalam daftar
3      if ((index + 1) % 5 == 0) {
4          //kondisi yang memeriksa apakah indeks item saat ini adalah kelipatan 5 (jika sisa dari (index + 1) dibagi oleh 5 adalah 0),
5          //yang menunjukkan bahwa pemisah khusus harus ditambahkan setelah item ke-5, ke-10, ke-15, dan seterusnya dalam daftar.
6          return Column(
7              //digunakan untuk mengembalikan widget
8              children: [
9                  //digunakan untuk menambahkan sejumlah widget sebagai anak-anak (children)
10                 SizedBox(height: 8), // Jarak di atas pemisah
11                 Divider(), //menambahkan garis pemisah (divider)
12                 SizedBox(height: 8), // Jarak di bawah pemisah
13                 Container( //sebagai pemisah antara item-item dalam daftar bunga
14                     color: Colors.orange, //mengatur latar belakang oranye dan berisi teks promosi.
15                     child: Center(
16                         //digunakan untuk memposisikan isi dari Container ke tengah (center)
17                         //secara horizontal dan vertikal dalam konteks pemisah.
18                         child: ListTile( //digunakan untuk menambahkan widget

```



```
1 title: Align( //menambahakan judul atau text
2             alignment: Alignment.center, //textnya agar berada di tengah
3             child: Text( //menambahakan text
4               'Dapatkan Potongan 5%', //memanggil tulisan ini agar tampil dilayar
5               style: TextStyle( //mengatur tampilan teks
6                 color: Colors.white, //mengatur warna
7                 fontSize: 20.0, //mengatur besar kecil font
8               ),
9             ),
10          ),
11        ),
12      ),
13    ],
14  ],
15 );
```



```
1 } else {
2   return Divider(color: Colors.transparent);
3 }
4 //Bagian ini adalah bagian dari fungsi `separatorBuilder` yang mengembalikan `Divider`
5 //dengan warna transparan jika kondisi `(index + 1) % 5 == 0` tidak terpenuhi.
6 //Ini menghasilkan pemisah transparan jika indeks item tidak merupakan kelipatan 5 dalam daftar bunga.
7 },
8 );
9 }
10 }
11
```

## Hasil Script Perbaikan Serta Penjelasan Setiap Line Yang Lebih Detail :

```
import 'package:flutter/material.dart';
//Mengimport library Flutter yang diperlukan

void main() {
  //Fungsi main() yang menjalankan aplikasi
  runApp(MyApp());
}
//runApp() dengan widget MyApp() sebagai root aplikasi.

class MyApp extends StatelessWidget {
  //Kelas MyApp tipe StatelessWidget.berarti MyApp adalah widget yang tidak
  dapat diubah setelah dibuat.
  @override
  //override menandakan bahwa metode di kelas anak menggantikan metode dengan
  nama yang sama di kelas induk atau antarmuka.
  Widget build(BuildContext context) {
    // untuk membangun tampilan dan mengembalikan widget yang akan ditampilkan
    pada layar.
    return MaterialApp(
      //digunakan sebagai akar (root) aplikasi Flutter untuk menyediakan fitur dan
      tampilan berdasarkan Material Design.
      home: Scaffold(
        //sebagai kerangka dasar yang berisi elemen-elemen seperti AppBar dan body.
        appBar: AppBar(
          //mengkonfigurasi bilah atas (appBar) dalam widget
          backgroundColor: Colors.orange,
          //mengatur warna latar belakang dari AppBar menjadi oranye.
          title: Text('Katalog Bunga - Diah Munica Nawang'),
          //mengatur teks judul yang akan ditampilkan di AppBar menjadi "Katalog Bunga
          - Diah Munica Nawang".
          leading: Icon(Icons.home),
          //menambahkan ikon rumah (home) sebagai elemen "leading" di dalam AppBar.
        ),
        body: FlowerList(),
        //daftar bunga yang telah didefinisikan akan ditampilkan sebagai konten
        utama dalam aplikasi.
      ),
    );
  }
}

class FlowerList extends StatelessWidget {
  //digunakan untuk membuat daftar bunga dalam aplikasi Flutter.
  final List<Map> bunga = [
    //sebagai daftar (List) yang berisi map (key-value pairs) yang
    merepresentasikan data bunga dalam aplikasi.
    {"nama": "Matahari", "harga": "65000", "gambar": "1.jpg"},
    {"nama": "Rose", "harga": "55000", "gambar": "2.jpg"},
    {"nama": "Dahlia", "harga": "20000", "gambar": "3.jpg"},
    {"nama": "Aster", "harga": "25000", "gambar": "4.jpg"},
    {"nama": "Krisan", "harga": "35000", "gambar": "5.jpg"},
    {"nama": "Lily", "harga": "40000", "gambar": "6.jpg"},
    {"nama": "Matahari", "harga": "35000", "gambar": "7.jpg"},
  ]
}
```



```

        {"nama": "Jakaranda", "harga": "40000", "gambar": "8.jpg"},
    ];
    //Baris-baris ini mendefinisikan beberapa data bunga dengan properti nama,
    harga, dan nama
    //berkas gambar yang akan digunakan dalam aplikasi.

    @override
    //digunakan untuk menunjukkan bahwa metode ini menggantikan metode yang
    sudah ada dalam kelas induk atau antarmuka yang sama
    Widget build(BuildContext context) {
    //membangun tampilan widget dan harus mengembalikan widget yang akan
    ditampilkan
        return ListView.separated(
    //daftar item terpisah dengan pemisah yang dapat dikonfigurasi di antara
    setiap item.
            padding: EdgeInsets.all(10),
    //mengatur jarak (padding) sekitar seluruh konten dalam
            itemCount: bunga.length,
    //mengatur jumlah item yang akan ditampilkan dalam
            itemBuilder: (context, index) {
    //fungsi yang digunakan untuk membangun setiap item dalam daftar
                final nama = bunga[index]["nama"];
    //mengambil nilai dari properti "nama" dari objek bunga yang sedang diproses
    dalam indeks
                final harga = bunga[index]["harga"];
    //mengambil nilai dari properti "harga" dari objek bunga yang sedang
    diproses dalam indeks
                final gambar = 'assets/images/${bunga[index]["gambar"]}';
    // menggabungkan teks 'assets/images/' dengan nilai dari properti "gambar"
    //dari objek bunga yang sedang diproses dalam indeks

                return Container(
    //gunakan untuk mengembalikan widget Container yang akan mengelilingi
    //dan memuat konten dari setiap item dalam daftar
                    height: 80, //mengatur tinggi (height) dari widget
                    decoration: BoxDecoration( //digunakan untuk mengatur dekorasi
    (style)
                        borderRadius: BorderRadius.all(Radius.circular(20)), //mengatur
    sudut (radius) dari border (pinggiran)
                        color: Colors.white, //mengatur warna
                        boxShadow: [ //digunakan untuk menambahkan bayangan (shadow)
                            BoxShadow( //mengatur warna, sebaran, kekaburan, dan offset
    bayangan.
                                color: Colors.grey.withOpacity(0.5), // Warna shadow
                                spreadRadius: 2, // Seberapa besar shadow
                                blurRadius: 4, // Seberapa kabur shadow
                                offset: Offset(0, 3), // Posisi shadow
                            ),
                        ],
                    ),
                    child: ListTile(
    //untuk menampilkan konten item dalam daftar dengan tata letak dan
    elemen yang telah ditentukan.
                        leading: Image.asset(gambar),
    //mengatur gambar yang akan ditampilkan
                        title: Column(

```

```

        //digunakan untuk menambahkan widget Column sebagai judul (title)
        dalam ListTile
        crossAxisAlignment: CrossAxisAlignment.start,
        //mengatur tata letak (layout) dari widget dalam

        children: [
            //digunakan untuk menambahkan sejumlah widget sebagai anak-anak
            (children)
            Text(nama), //menambahkan teks yang akan ditampilkan
            Row( //digunakan untuk membuat baris horizontal
                mainAxisAlignment: MainAxisAlignment.min, //mengatur ukuran utama
                (main axis) dari Row agar sesempit mungkin,
                children: [ //digunakan untuk menambahkan sejumlah widget
                sebagai anak-anak (children) dalam Row,
                    Icon(Icons.star, color: Colors.yellow),
                    Icon(Icons.star, color: Colors.yellow),
                    Icon(Icons.star, color: Colors.yellow),
                    Icon(Icons.star, color: Color.fromARGB(255, 87, 85,
74))),
                    Icon(Icons.star, color: Color.fromARGB(255, 87, 85,
74))),
                //Bagian ini menambahkan beberapa ikon bintang ke dalam `Row`
                sebagai anak-anak (children) dalam baris horizontal.
                //Ikon-ikon ini digunakan untuk menampilkan peringkat item dalam
                bentuk bintang, dengan beberapa ikon bintang diwarnai kuning (yellow)
                //dan beberapa ikon bintang diwarnai sesuai dengan kode warna
                RGB yang ditentukan.
                ],
            ),
            Text('Harga: $harga'),
            //menampilkan teks "Harga:" diikuti oleh nilai harga yang
            diambil dari variabel harga.
            ],
        ),
    ),
);
},
separatorBuilder: (context, index) {
    // digunakan untuk membangun pemisah antara item-item dalam daftar
    if ((index + 1) % 5 == 0) {
        //kondisi yang memeriksa apakah indeks item saat ini adalah kelipatan
        5 (jika sisa dari (index + 1) dibagi oleh 5 adalah 0),
        //yang menunjukkan bahwa pemisah khusus harus ditambahkan setelah item
        ke-5, ke-10, ke-15, dan seterusnya dalam daftar.
        return Column(
            //digunakan untuk mengembalikan widget
            children: [
                //digunakan untuk menambahkan sejumlah widget sebagai anak-anak
                (children)
                SizedBox(height: 8), // Jarak di atas pemisah
                Divider(), //menambahkan garis pemisah (divider)
                SizedBox(height: 8), // Jarak di bawah pemisah
                Container( //sebagai pemisah antara item-item dalam daftar
                bunga
                    color: Colors.orange, //mengatur latar belakang oranye dan
                    berisi teks promosi.
                    child: Center(

```

```

        //digunakan untuk memposisikan isi dari Container ke tengah (center)
        //secara horizontal dan vertikal dalam konteks pemisah.
        child: ListTile( //digunakan untuk menambahkan widget
            title: Align( //menambahakan judul atau text
                alignment: Alignment.center, //textnya agar berada di
tengah

                child: Text( //menambahkan text
                    'Dapatkan Potongan 5%', //memanggil tulisan ini agar
tampil dilayar

                    style: TextStyle( //mengatur tampilan teks
                        color: Colors.white, //mengatur warna
                        fontSize: 20.0, //mengatur besar kecil font
                    ),
                ),
            ),
        ),
    ),
],
);
} else {
    return Divider(color: Colors.transparent);
}

//Bagian ini adalah bagian dari fungsi `separatorBuilder` yang
mengembalikan `Divider`
//dengan warna transparan jika kondisi `(index + 1) % 5 == 0` tidak
terpenuhi.
//Ini menghasilkan pemisah transparan jika indeks item tidak merupakan
kelipatan 5 dalam daftar bunga.
    },
);
}
}

```