	LAPORAN PRAKTIKUM	PERTEMUAN 8
	PEMROGRAMAN PERANGKAT BERGERAK	
	TUGAS FLUTTER UI	

A. IDENTITAS

NIM : 2103067
 Nama Lengkap : FATWAH FIRMANSYAH SLAMET RIYADI
 Kelas : D3TI2C
 Program Studi : D3 Teknik Informatika
 Jurusan : Teknik Informatika

B. WAKTU

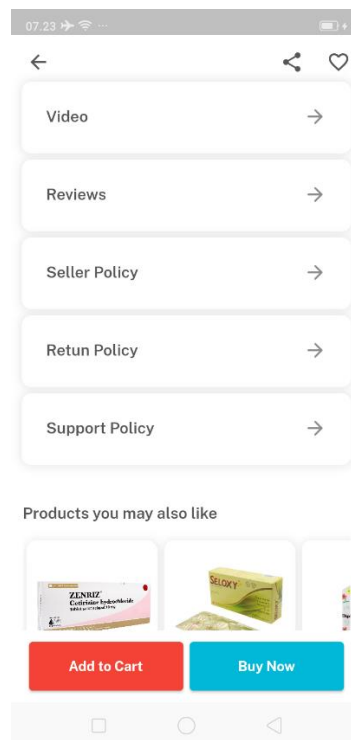
Tanggal : 12 April 2023
 Waktu : 10.00 s/d 11.40

C. TUGAS

Untuk pengerjaan tugas kali ini saya akan mengambil sample dari proyek 3 saya.

1. Membuat Tampilan dengan ListView

Pada poin pertama, saya menerapkan penggunaan listview pada halaman detail produk berikut adalah tampilannya:



ListView tersebut saya gunakan sebagai tombol yang terkait dengan suatu product seperti review produk dan lain lain.

Untuk codenya dapat dilihat pada gambar dibawah ini.



Pada code diatas saya menggunakan listview kemudian saya generate sebuah list dengan panjang dari sebuah list yang berisi judul dari tiap tombol yang sudah saya buat sebelumnya. Kemudian didalam list generate tersebut akan mengembalikan sebuah container yang berisi ListTile, yang isinya adalah judul untuk tombolnya dan trailing berupa icon panah kanan.

2. Membuat Tampilan dengan GridView

Untuk GridView saya terapkan untuk menampilkan data produk dari suatu kategori yang ada. Berikut ini adalah tampilan screen product:



Untuk code dari gambar diatas dapat dilihat pada gambar dibawah ini:

```

1 class CategoryDetailView extends GetView {
2   final String? title;
3   const CategoryDetailView({Key? key, required this.title}) : super(key: key);
4   @override
5   Widget build(BuildContext context) {
6     Get.lazyPut(() => ProductController());
7     var productC = Get.find<ProductController>();
8     return bgWidget(
9       child: Scaffold(
10        appBar: AppBar(
11          title: title!.text.fontFamily(bold).size(18).white.make(),
12          elevation: 0,
13        ),
14        body: Column(
15          children: [
16            Container(
17              padding: const EdgeInsets.all(8),
18              child: SingleChildScrollView(
19                scrollDirection: Axis.horizontal,
20                physics: const BouncingScrollPhysics(),
21                child: Row(
22                  children: List.generate(
23                    6,
24                    (index) => Container(
25                      margin: const EdgeInsets.only(right: 6),
26                      decoration: BoxDecoration(
27                        borderRadius: BorderRadius.circular(8),
28                        color: whiteColor),
29                      width: Get.width * 0.35,
30                      height: Get.height * 0.07,
31                      child: const Align(
32                        alignment: Alignment.center,
33                        child: Text('Category')),
34                    )),
35              ),
36            ),
37          ],
38          Expanded(
39            child: Container(
40              padding: const EdgeInsets.only(top: 10, right: 10, left: 10),
41              color: light,
42              child: GridView.builder(
43                physics: const BouncingScrollPhysics(),
44                shrinkWrap: true,
45                itemCount: 6,
46                gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
47                  crossAxisCount: 2,
48                  mainAxisExtent: Get.height * 0.30,
49                  mainAxisSpacing: 10,
50                  crossAxisSpacing: 8),
51                itemBuilder: (context, index) {
52                  return Product(
53                    image: featuredPImg[index],
54                    title: featuredPTitle[index],
55                    price: featuredPPPrice[index],
56                  ).onTap(() {
57                    productC.title = featuredPTitle[index];
58                    productC.price = featuredPPPrice[index];
59                    Get.toNamed(Routes.PRODUCT);
60                  });
61                },
62              )),
63            ),
64          ],
65        ),
66      ));
67   }
68 }

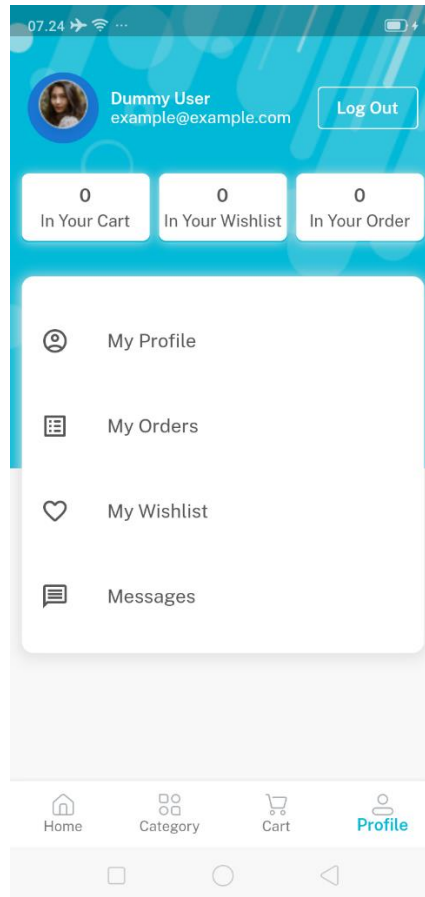
```

Pada source code diatas dapat dilihat bahwa saya menggunakan GridView builder dengan jumlah 6 data karena datanya masih berbentuk dummy data. Kemudian untuk item buildernya akan mengembalikan sebuah Widget yang sudah saya buat, dimana widget tersebut merupakan bentuk satuan dari produk (hasil extract widget). Berikut ini adalah isi dari widgetnya.

```
1 class Product extends StatelessWidget {
2   final String? image;
3   final String? title;
4   final String? price;
5   const Product({
6     super.key, this.image, this.title, this.price,
7   });
8
9   @override
10  Widget build(BuildContext context) {
11    return Container(
12      height: Get.height * 0.28,
13      width: Get.width * 0.37,
14      decoration: BoxDecoration(
15        color: whiteColor,
16        borderRadius: BorderRadius.circular(10),
17        boxShadow: [
18          BoxShadow(
19            blurRadius: 8,
20            color: Colors.grey.shade300,
21            spreadRadius: 0.5,
22            blurStyle: BlurStyle.outer,
23            offset: const Offset(1, 2))
24        ],
25      ),
26      child: LayoutBuilder(builder: (context, constraints) {
27        return Column(
28          crossAxisAlignment: CrossAxisAlignment.start,
29          children: [
30            ClipRRect(
31              borderRadius: const BorderRadius.only(
32                topLeft: Radius.circular(10), topRight: Radius.circular(10)),
33              child: Image.asset(image!, width: constraints.maxWidth),
34            ),
35            5.heightBox,
36            title!
37              .text
38              .fontFamily(semibold)
39              .color(darkFontGrey)
40              .make()
41              .marginOnly(left: 10),
42            5.heightBox,
43            price!
44              .text
45              .fontFamily(semibold)
46              .color(blue)
47              .size(14)
48              .make()
49              .marginOnly(left: 10),
50          ],
51        );
52      }),
53    );
54  }
55 }
```

3. Membuat Profile Screen

Untuk profile screen saya menggunakan stack dan positioned. Hasilnya dapat dilihat pada gambar berikut:



Pada profile screen tersebut terdapat beberapa elemen, diantaranya adalah circle avatar, container untuk menampilkan data tertentu yang berfungsi sebagai tombol juga, serta beberapa tombol yang saya generate menggunakan listview. Source code dapat dilihat pada gambar dibawah ini.

```

1 class ProfileView extends GetView<ProfileController> {
2   const ProfileView({Key? key}) : super(key: key);
3   @override
4   Widget build(BuildContext context) {
5     return Container(
6       child: Stack(
7         children: [
8           Container(
9             child: Column(
10              children: [
11                SizedBox(
12                  height: Get.height * 0.55,
13                  width: Get.width,
14                  child: Image.asset(
15                    imgBg,
16                    fit: BoxFit.cover,
17                  ),
18                ),
19                Container(
20                  height: Get.height * 0.37134,
21                  width: Get.width,
22                  color: light,
23                )
24              ],
25            ),
26          ),
27          Positioned(
28            left: 15,
29            right: 15,
30            top: Get.height * 0.08,
31            child: Container(
32              width: Get.width,
33              height: 60,
34              child: Row(
35                crossAxisAlignment: CrossAxisAlignment.center,
36                children: [
37                  const CircleAvatar(
38                    radius: 30,
39                    backgroundImage: AssetImage(imgProfile),
40                  ),
41                  Expanded(
42                    child: Column(
43                      mainAxisAlignment: MainAxisAlignment.center,
44                      crossAxisAlignment: CrossAxisAlignment.start,
45                      children: [
46                        "Dummy User".text.fontFamily(semibold).white.make(),
47                        "example@example.com".text.white.make()
48                      ],
49                    ).marginOnly(left: 10),
50                  ),
51                  OutlinedButton(
52                    style: OutlinedButton.styleFrom(
53                      side: const BorderSide(color: whiteColor)),
54                    onPressed: () {},
55                    child: "Log Out".text.white.fontFamily(semibold).make(),
56                  )
57                ],
58              ),
59            ),
60          ),

```

```

27     Positioned(
28       left: 15,
29       right: 15,
30       top: Get.height * 0.08,
31       child: Container(
32         width: Get.width,
33         height: 60,
34         child: Row(
35           crossAxisAlignment: CrossAxisAlignment.center,
36           children: [
37             const CircleAvatar(
38               radius: 30,
39               backgroundImage: AssetImage(imgProfile),
40             ),
41             Expanded(
42               child: Column(
43                 mainAxisAlignment: MainAxisAlignment.center,
44                 crossAxisAlignment: CrossAxisAlignment.start,
45                 children: [
46                   "Dummy User".text.fontFamily(semibold).white.make(),
47                   "example@example.com".text.white.make()
48                 ],
49               ).marginOnly(left: 10),
50             ),
51             OutlinedButton(
52               style: OutlinedButton.styleFrom(
53                 side: const BorderSide(color: whiteColor)),
54               onPressed: () {},
55               child: "Log Out".text.white.fontFamily(semibold).make(),
56             )
57           ],
58         ),
59       ),
60     ),
61     Positioned(
62       left: 10,
63       right: 10,
64       top: Get.height * 0.2,
65       child: Column(
66         children: [
67           Row(
68             mainAxisAlignment: MainAxisAlignment.spaceBetween,
69             children: List.generate(
70               3,
71               (index) => ProfileButton(
72                 height: Get.height * 0.08,
73                 width: Get.width / 3.33,
74                 title: index == 0
75                   ? "In Your Cart"
76                   : index == 1
77                     ? "In Your Wishlist"
78                     : "In Your Order",
79                 value: index == 0
80                   ? "0"
81                   : index == 1
82                     ? "0"
83                     : "0",
84               )),
85           ),
86         ),
87       30.heightBox,
88       Container(
89         padding: const EdgeInsets.only(bottom: 20),
90         width: Get.width - 20,
91         decoration: BoxDecoration(
92           color: whiteColor,
93           borderRadius: BorderRadius.circular(10),
94           boxShadow: [
95             BoxShadow(
96               blurRadius: 10,
97               color: Colors.grey.shade300,
98               spreadRadius: 2,
99               offset: const Offset(0, 2))
100           ],
101       ),
102       child: ListView.separated(
103         shrinkWrap: true,
104         itemCount: profileButtonsList.length,
105         separatorBuilder: (context, index) => const Divider(color: whiteColor,),
106         itemBuilder: (context, index) => ListTile(
107           leading: profileButtonsIcon[index],
108           title: profileButtonsList[index].text.color(darkFontGrey).make(),
109           onTap: () {},
110         ),
111       ),
112     ),
113   ],
114 ),
115 ),
116 ),
117 ),
118 );
119 }
120 }

```


Untuk tiga tombol diatas saya juga menggunakan widget yang sudah saya extract sebelumnya, berikut ini adalah widget untuk tombol yang sudah saya extract.

```
1 class ProfileButton extends StatelessWidget {
2   final double height;
3   final double width;
4   final String value;
5   final String title;
6
7   const ProfileButton({
8     super.key,
9     required this.height,
10    required this.width,
11    required this.value,
12    required this.title,
13  });
14
15  @override
16  Widget build(BuildContext context) {
17    return Container(
18      height: height,
19      width: width,
20      decoration: BoxDecoration(
21        color: Colors.white,
22        boxShadow: [
23          BoxShadow(
24            blurRadius: 7,
25            color: Colors.grey.shade300,
26            spreadRadius: 0.3,
27            blurStyle: BlurStyle.outer,
28            offset: const Offset(0, 1))
29        ],
30      borderRadius: BorderRadius.circular(6),
31    ),
32    child: Column(
33      mainAxisAlignment: MainAxisAlignment.center,
34      children: [
35        value.text.fontFamily(semibold).size(16).color(darkFontGrey).make(),
36        4.heightBox,
37        title.text.color(darkFontGrey).make(),
38      ],
39    ),
40  );
41 }
42 }
```