

Nama : Akmal Muhamad Firdaus

NIM : 1301204188

## Jurnal Praktikum Modul 11

### 1. Packages

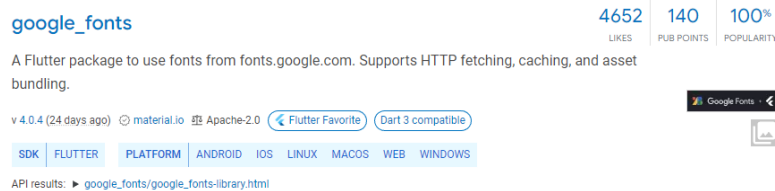
#### 1.1. Pengenalan Packages

Packages pada Flutter adalah kumpulan kode yang telah dibuat oleh pengembang lain untuk membantu dalam pengembangan aplikasi Flutter. Package dapat berisi fungsi dan komponen yang siap digunakan seperti widget, utilitas, plugin, atau alat bantu lainnya. Dengan menggunakan package, pengembang dapat mengimpor kode yang telah dikembangkan sebelumnya dan menggunakannya dalam aplikasi mereka tanpa perlu membuat semuanya dari awal. Packages dapat diunduh melalui situs web Flutter atau melalui pengelola paket seperti pub.dev, dan dapat disertakan dalam proyek Flutter dengan menggunakan dependensi yang sesuai dalam file pubspec.yaml.

#### 1.2. Penggunaan Packages

Penggunaan packages dalam Flutter melibatkan beberapa langkah berikut:

1. Temukan package yang ingin kita gunakan. Kita dapat mencarinya di situs web Flutter seperti pub.dev atau menggunakan pencarian di dalam editor Flutter seperti Visual Studio Code.



2. Buka file pubspec.yaml dalam proyek Flutter. File ini berfungsi sebagai manifest proyek dan berisi daftar dependensi.
4. Di bagian dependencies dalam file pubspec.yaml, tambahkan entri untuk package yang ingin kita gunakan. Contohnya seperti ini:

```
#Akmal Muhamad Firdaus 1301204188
dependencies:
  flutter:
    sdk: flutter
  google_fonts: ^4.0.4
```

5. Save file tersebut lalu jalan kan perintah “flutter packages get” pada cmd

```
C:\xampp\htdocs\ASP\modul11>flutter packages get
Waiting for another Flutter command to release the startup lock...
Running "flutter pub get" in modul11...
Resolving dependencies... (1.6s)
  async 2.10.0 (2.11.0 available)
  characters 1.2.1 (1.3.0 available)
  collection 1.17.0 (1.17.2 available)
  crypto 3.0.3
  ffi 2.0.2
  file 6.1.4 (7.0.0 available)
  google_fonts 4.0.4
  http 0.13.6
  http_parser 4.0.2
  js 0.6.5 (0.6.7 available)
  lints 2.0.1 (2.1.0 available)
  matcher 0.12.13 (0.12.16 available)
  material_color_utilities 0.2.0 (0.5.0 available)
  meta 1.8.0 (1.9.1 available)
  path 1.8.2 (1.8.3 available)
  path_provider 2.0.15
  path_provider_android 2.0.27
  path_provider_foundation 2.2.3
  path_provider_linux 2.1.10
  path_provider_platform_interface 2.0.6
  path_provider_windows 2.1.6
  platform 3.1.0
  plugin_platform_interface 2.1.4
  process 4.2.4
  source_span 1.9.1 (1.10.0 available)
  test_api 0.4.16 (0.6.0 available)
  typed_data 1.3.2
  win32 4.1.4 (5.0.2 available)
  xdg_directories 1.0.4
Changed 18 dependencies!
```

## 2. User Interaction

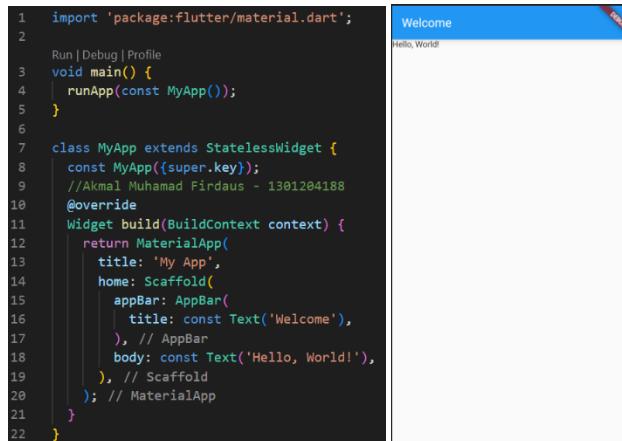
### 2.1. Stateful & Stateless

Terdapat dua jenis utama dari widget yang digunakan untuk membangun antarmuka pengguna: Stateful Widget dan Stateless Widget.

#### 2.1.1. Stateless Widget

Stateless widget adalah widget yang tidak memiliki state internal yang berubah. Artinya, setelah dibuat, properti atau tampilan widget tersebut tidak dapat diubah. Stateless widget cocok digunakan untuk bagian tampilan statis yang tidak memerlukan perubahan.

Contoh penggunaan Stateless Widget:



#### 2.1.2. Statefull Widget

Stateful widget adalah widget yang memiliki state internal yang dapat berubah seiring waktu. Stateful widget cocok digunakan ketika ada bagian tampilan yang perlu diperbarui secara dinamis, misalnya ketika ada interaksi pengguna atau perubahan data.

Contoh penggunaan Stateful Widget:



## 2.2. Form

Form digunakan untuk mengumpulkan input dari pengguna menggunakan widget Form dan TextFormField. Berikut adalah contoh sederhana untuk membuat formulir di Flutter:

```
class MyForm extends StatefulWidget {  
  @override  
  _MyFormState createState() => _MyFormState();  
}
```

```
class _MyFormState extends State<MyForm> {  
  final _formKey = GlobalKey<FormState>();  
  late String _name;  
  late String _email;  
  
  void _submitForm() {  
    if (_formKey.currentState?.validate() == true) {  
      _formKey.currentState?.save();  
      print('Name: $_name');  
      print('Email: $_email');  
    }  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text('Akmal Muhamad Firdaus - 1301204188'),  
      ),  
      body: Padding(  
        padding: EdgeInsets.all(16.0),  
        child: Form(  
          key: _formKey,  
          child: Column(  
            children: [  
              TextFormField(  
                decoration: InputDecoration(labelText: 'Name'),  
                validator: (value) {  
                  if (value == null || value.isEmpty) {  
                    return 'Please enter your name';  
                  }  
                  return null;  
                },  
                onSaved: (value) {  
                  if (value != null) {  
                    _name = value;  
                  }  
                },  
              ),  
              TextFormField(  
                decoration: InputDecoration(labelText: 'Email'),  
                validator: (value) {  
                  if (value == null || value.isEmpty) {  
                    return 'Please enter your email';  
                  }  
                  return null;  
                },  
                onSaved: (value) {  
                  if (value != null) {  
                    _email = value;  
                  }  
                },  
              ),  
              SizedBox(height: 16.0),  
              ElevatedButton(  
                child: const Text('Submit'),  
                onPressed: _submitForm,  
              ),  
            ],  
          ),  
        ),  
      ),  
    );  
  }  
}
```

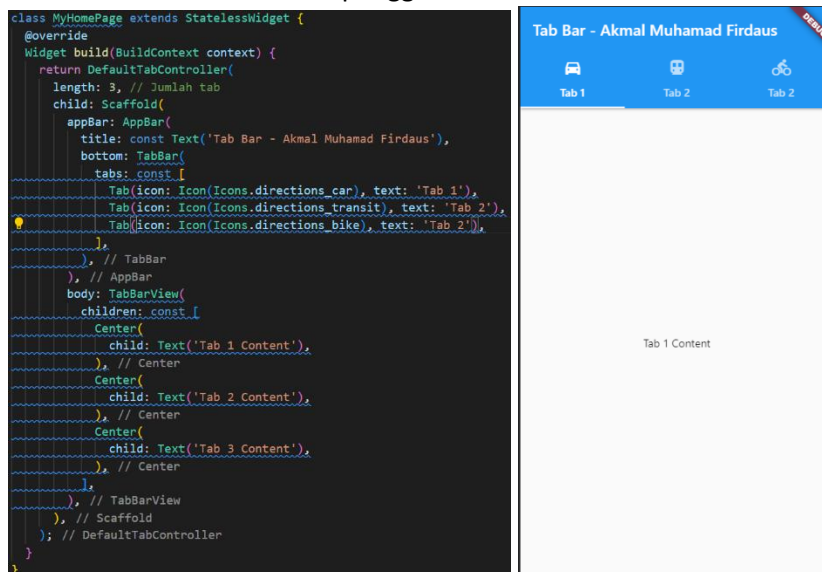
The screenshot shows the visual output of the Flutter code. It features a blue title bar with the text 'Akmal Muhamad Firdaus - 1301204188'. Below the title bar, there are two text input fields. The first field is labeled 'Name' and contains the text 'Akmal Muhamad Firdaus'. The second field is labeled 'Email' and contains the text 'akmalmf007@gmail.com'. At the bottom of the form, there is a blue 'Submit' button.

## 2.3. Menu

Salah satu hal penting dari pembuatan aplikasi adalah menu. Menu ini berfungsi untuk separasi antar fitur atau page. Sulit rasanya apabila semua fitur ditampilkan dalam satu halaman, selain sulit pengguna akan kesulitan dalam mengoperasikannya. Maka disini menu page sangat bermanfaat. Secara umum terdapat 2 jenis widget menu yang sering digunakan, yaitu `bottom navigation bar` dan `tab bar`. Karena Flutter mendukung penuh guideline yang dibuat oleh Google, yaitu Material Design.

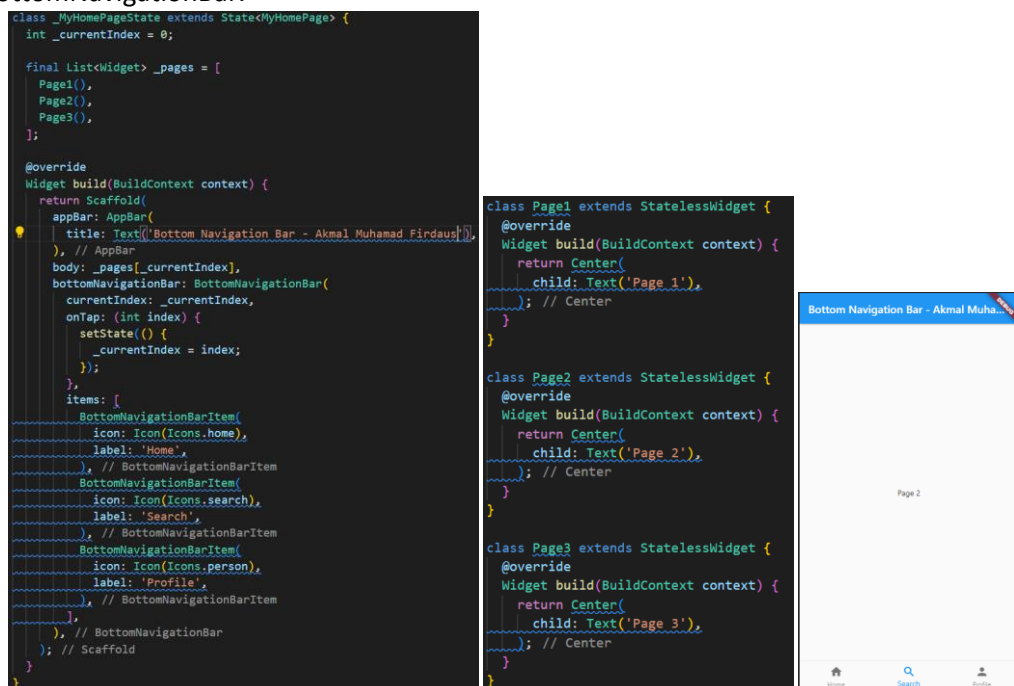
### 2.3.1. Tab Bar

Di Flutter, kita dapat menggunakan `TabBar` dan `TabBarView` untuk membuat tampilan dengan tab-bar yang memungkinkan pengguna beralih antara beberapa konten yang terkait. Berikut adalah contoh penggunaan `TabBar`:



### 2.3.2. Bottom Navigation Bar

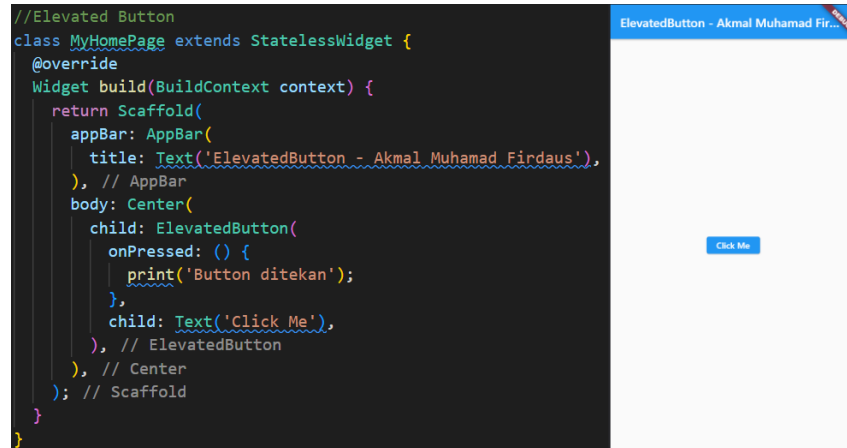
Di Flutter, kita dapat menggunakan `BottomNavigationBar` untuk membuat navigasi bawah (bottom navigation) yang memungkinkan pengguna beralih antara beberapa halaman atau tampilan yang terkait. Berikut adalah contoh penggunaan `BottomNavigationBar`:



## 2.4. Buttons

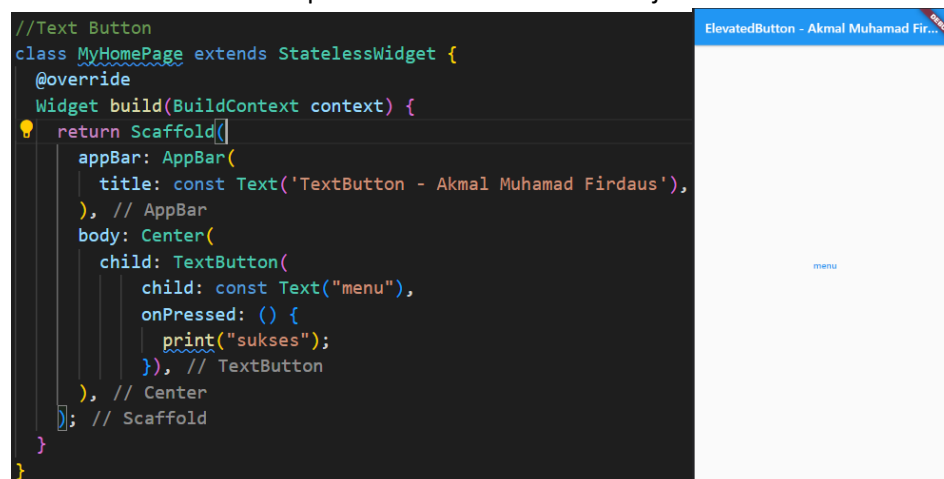
### 2.4.1. Elevated Button

ElevatedButton adalah sebuah widget yang digunakan untuk membuat tombol dengan tampilan raised atau menonjol. Tombol ini memiliki bayangan dan tampilan yang memberikan efek visual yang lebih menonjol dibandingkan dengan tombol datar.



### 2.4.2. Text Button

TextButton adalah sebuah widget yang digunakan untuk membuat tombol dengan tampilan teks. Berbeda dengan ElevatedButton yang memiliki tampilan raised, TextButton memiliki tampilan datar atau tidak menonjol.



### 2.4.3. DropdownButton

DropdownButton adalah sebuah widget yang digunakan untuk membuat dropdown atau menu pilihan yang dapat dipilih oleh pengguna. DropdownButton menampilkan daftar opsi atau item, dan pengguna dapat memilih satu opsi dari daftar tersebut.

```
//Dropdown Button
class MyHomePage extends StatefulWidget {
  @override
  _MyHomePageState createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  String? selectedOption;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('DropdownButton Akmal Muhamad Firdaus'),
      ), // AppBar
      body: Center(
        child: DropdownButton<String>({
          value: selectedOption,
          onChanged: (String? newValue) {
            setState(() {
              selectedOption = newValue;
            });
          },
          items: <String>[
            'Pilihan 1',
            'Pilihan 2',
            'Pilihan 3',
            'Pilihan 4',
          ].map<DropdownMenuItem<String>>((String value) { // <Str
            return DropdownMenuItem<String>({
              value: value,
              child: Text(value),
            }); // DropdownMenuItem
          }).toList(),
        ), // DropdownButton
      ), // Center
    ); // Scaffold
  }
}
```

