

**LAPORAN PRAKTIKUM**  
**MATA KULIAH PEMROGRAMAN MOBILE**



**U N I V E R S I T A S**  
**AMIKOM PURWOKERTO**

**Oleh :**

**NAMA: FAJAR RAMADHAN**

**NIM: 23SA31A065**

**PROGRAM STUDI TEKNOLOGI INFORMASI**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS AMIKOM**  
**PURWOKERTO**

**2025**

## PRAKTIKUM 6 MEMBUAT FORMULIR YANG MELIPUTI BERBAGAI WIDGET

### 1. Model.dart

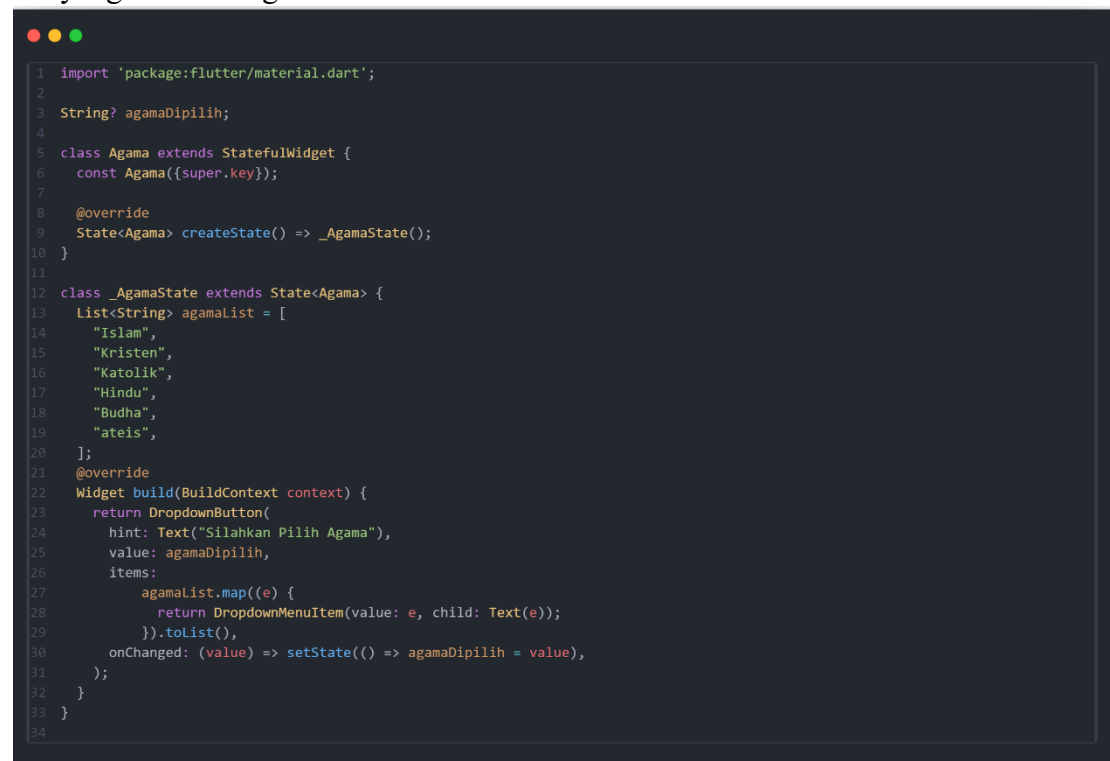
Pada file ini meliputi model untuk menampung keseluruhan data yang akan diinputkan melalui formulir yang dihasilkan, ini sangat penting untuk menjaga konsistensi struktur data yang ada. Didalamnya berupa class dengan berbagai properties yang diperlukan, dan akan di buat semacam penampungnya di generic list pendaftaranlist. Didalamnya juga ada method yang akan menampilkan pesan supaya formulir tidak ada yang tidak diisi

```
1 import 'package:flutter/material.dart';
2
3 class Pendaftaran {
4   BuildContext context;
5
6   String nama = "",
7     email = "",
8     noTelp = "",
9     jenisKelamin = "",
10    bahasa = "",
11    agama = "",
12    tanggalDaftar = "",
13    jamDaftar = "";
14
15   Pendaftaran(
16     this.context,
17     this.nama,
18     this.email,
19     this.noTelp,
20     this.jenisKelamin,
21     this.bahasa,
22     this.agama,
23     this.tanggalDaftar,
24     this.jamDaftar,
25   );
26
27   bool sudahValid() {
28     if (nama.isEmpty) {
29       tampilkanPesan("nama tidak boleh kosong 🚫");
30       return false;
31     } else if (noTelp.isEmpty) {
32       tampilkanPesan("no telp tidak boleh kosong 🚫");
33       return false;
34     } else if (email.isEmpty) {
35       tampilkanPesan("email tidak boleh kosong 🚫");
36       return false;
37     } else if (jenisKelamin.isEmpty) {
38       tampilkanPesan("jenis kelamin tidak boleh kosong 🚫");
39       return false;
40     } else if (bahasa.isEmpty) {
41       tampilkanPesan("bahasa tidak boleh kosong 🚫");
42       return false;
43     } else if (agama.isEmpty) {
44       tampilkanPesan("agama tidak boleh kosong 🚫");
45       return false;
46     } else if (tanggalDaftar.isEmpty) {
47       tampilkanPesan("tanggal daftar tidak boleh kosong 🚫");
48       return false;
49     } else if (jamDaftar.isEmpty) {
50       tampilkanPesan("jam daftar tidak boleh kosong 🚫");
51       return false;
52     }
53     return true;
54   }
55
56   void tampilkanPesan(String pesan) {
57     ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(pesan)));
58   }
59 }
60
61 List<Pendaftaran> pendaftaranList = [];
```

Gambar 1. Kode model.dart

## 2. Agama.dart

Pada file ini berisi statefullwidget untuk formulir dapat memilih list agama dalam bentuk widget DropDownButton, item pada dropdown merupakan hasil mapping collection pada list yang bernama agamaList



```
1 import 'package:flutter/material.dart';
2
3 String? agamaDipilih;
4
5 class Agama extends StatefulWidget {
6   const Agama({super.key});
7
8   @override
9   State<Agama> createState() => _AgamaState();
10 }
11
12 class _AgamaState extends State<Agama> {
13   List<String> agamaList = [
14     "Islam",
15     "Kristen",
16     "Katolik",
17     "Hindu",
18     "Budha",
19     "ateis",
20   ];
21   @override
22   Widget build(BuildContext context) {
23     return DropdownButton(
24       hint: Text("Silahkan Pilih Agama"),
25       value: agamaDipilih,
26       items:
27         agamaList.map((e) {
28           return DropdownMenuItem(value: e, child: Text(e));
29         }).toList(),
30       onChanged: (value) => setState(() => agamaDipilih = value),
31     );
32   }
33 }
34
```

Gambar 2. Kode agama.dart

## 3. kemampuanBahasa.dart

Pada file ini berisi statefull widget untuk pilihan bahasa di formulir, disini mengimplementasikan widget Gridview yang mapping collection pilihan bahasa pada list bahasa dan menerapkan extracted widget OpsiBahasa yang memiliki properties strBahasa sebagai bahasa yang dipilih dan akan ditampung jika value checkbox pada widget tersebut bernilai true ke bahasaDipilihList yang di assign di awal kode yang nanti akan dikirim sebagai properties di model.dart.

```

1 import 'package:flutter/material.dart';
2
3 List<String> bahasaDipilihList = [];
4
5 class Kemampuanbahasa extends StatefulWidget {
6   const Kemampuanbahasa({super.key});
7
8   @override
9   State<Kemampuanbahasa> createState() => _KemampuanbahasaState();
10 }
11
12 class _KemampuanbahasaState extends State<Kemampuanbahasa> {
13   List<String> bahasaList = [
14     "Indonesia",
15     "Inggris",
16     "Jawa",
17     "Sunda",
18     "Batak",
19     "Madura",
20     "Jepang",
21     "Korea",
22     "Mandarin",
23   ];
24   @override
25   Widget build(BuildContext context) {
26     return Column(
27       crossAxisAlignment: CrossAxisAlignment.start,
28       children: [
29         Text("Kemampuan Berbahasa"),
30         SizedBox(
31           height: 90,
32           child: GridView(
33             gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
34               crossAxisCount: 2,
35               childAspectRatio: 4 / 1,
36             ),
37             children: bahasaList.map((e) => OpsiBahasa(strBahasa: e)).toList(),
38           ),
39         ],
40     );
41   }
42 }
43
44 class OpsiBahasa extends StatefulWidget {
45   final String strBahasa;
46   const OpsiBahasa({super.key, required this.strBahasa});
47
48   @override
49   State<OpsiBahasa> createState() => _OpsiBahasaState();
50 }
51
52 class _OpsiBahasaState extends State<OpsiBahasa> {
53   bool? opsiDipilih = false;
54   @override
55   Widget build(BuildContext context) {
56     return Row(
57       children: [
58         Checkbox(
59           value: opsiDipilih,
60           onChanged: (value) {
61             setState(() {
62               opsiDipilih = value;
63               if (value == true) {
64                 bahasaDipilihList.add(widget.strBahasa);
65               } else {
66                 bahasaDipilihList.remove(widget.strBahasa);
67               }
68             });
69           },
70         ),
71         Text(widget.strBahasa),
72       ],
73     );
74   }
75 }
76 }
77

```

Gambar 3. Kode pilihanBahasa.dart

#### 4. pilihTanggal.dart

Pada kode di file ini berisi textfield yang mengatur pilihan tanggal untuk properties pada model. Pilihan tanggal ini dibuat dalam method pilihanTanggal dengan tipe Future yang dimana artinya berupa asynchronus kode yang menunggu data dari method DateTimePicker diisi baru akan diproses kodenya. DateTimePicker merupakan method bawaan flutter yang akan menampilkan dialog pemilihan tanggal.

```
1 import 'package:flutter/material.dart';
2
3 TextEditingController tanggalDaftarDaftarController = TextEditingController();
4
5 class Tanggaldaftar extends StatefulWidget {
6   const Tanggaldaftar({super.key});
7
8   @override
9   State<Tanggaldaftar> createState() => _TanggaldaftarState();
10 }
11
12 class _TanggaldaftarState extends State<Tanggaldaftar> {
13   @override
14   Widget build(BuildContext context) {
15     return TextField(
16       controller: tanggalDaftarDaftarController,
17       decoration: InputDecoration(
18         label: Text("Tanggal Daftar"),
19         suffixIcon: IconButton(
20           onPressed: () {
21             setState(() {
22               pilihTanggal();
23             });
24           },
25           icon: Icon(Icons.date_range),
26         ),
27       ),
28     );
29   }
30
31   Future<void> pilihTanggal() async {
32     DateTime? tglDidapat;
33
34     tglDidapat = await showDatePicker(
35       context: context,
36       firstDate: DateTime(2025),
37       lastDate: DateTime(2050),
38     );
39
40     if (tglDidapat.toString() != "null") {
41       tanggalDaftarDaftarController.text = tglDidapat.toString().substring(
42         0,
43         10,
44       );
45     }
46   }
47 }
48
```

Gambar 4. Kode pilihTanggal.dart

#### 5. jamDaftar.dart

Hampir sama seperti file pilihTanggal.dart di dalam jamDaftar.dart berisi widget textfield yang menampung hasil dari dateTimePicker namun dengan format waktu pada hari ini. Bahkan nama method untuk menghandlenya dibuat sama dengan nama pilihanTanggal dan future method yang bersifat asynchronus

```

1 // ignore_for_file: file_names
2
3 import 'package:flutter/material.dart';
4
5 TextEditingController jamDaftarDaftarController = TextEditingController();
6
7 class Jamdaftar extends StatefulWidget {
8   const Jamdaftar({super.key});
9
10  @override
11  State<Jamdaftar> createState() => _JamdaftarState();
12 }
13
14 class _JamdaftarState extends State<Jamdaftar> {
15   @override
16   Widget build(BuildContext context) {
17     return TextField(
18       controller: jamDaftarDaftarController,
19       decoration: InputDecoration(
20         label: Text("Jam Daftar"),
21         suffix: IconButton(
22           onPressed: () {
23             pilihTanggal();
24           },
25           icon: Icon(Icons.timer),
26         ),
27       ),
28     );
29   }
30
31   Future<void> pilihTanggal() async {
32     TimeOfDay? tglDidapat;
33     tglDidapat = await showTimePicker(
34       context: context,
35       initialTime: TimeOfDay.now(),
36     );
37     if (tglDidapat.toString() != "null") {
38       jamDaftarDaftarController.text = tglDidapat
39         .toString()
40         .replaceAll('TimeOfDay(', '')
41         .replaceAll(')', '');
42     }
43   }
44 }
45

```

Gambar 5. Kode jamDaftar.dart

#### 6. dataPendaftaran.dart

Pada file ini berisi kode yang akan menampilkan keseluruhan data yang ada dan telah ditambahkan ke dalam model.dart, dimana di kode program terlihat jelas di bagian atas mengimport mode.dart, yang dimana data pada model yang ditampung oleh list penampung listPendaftaran di model.dart di tampilkan dengan widget listBuilder. Kemudian didalamnya juga terdapat widget showDialog untuk konfirmasi hapus salah satu item list dengan memanfaatkan method list dart removeAt()

```

1 // ignore: file_names
2 import 'package:flutter/material.dart';
3 import 'package:praktikum_6/model.dart';
4
5 class Datapendaftaran extends StatefulWidget {
6   const Datapendaftaran({super.key});
7
8   @override
9   State<Datapendaftaran> createState() => _DatapendaftaranState();
10 }
11
12 class _DatapendaftaranState extends State<Datapendaftaran> {
13   @override
14   Widget build(BuildContext context) {
15     return Scaffold(
16       appBar: AppBar(title: Text("Data Pendaftaran")),
17       body: ListView.builder(
18         itemCount: pendaftaranList.length,
19         itemBuilder:
20           (context, index) => Card(
21             child: ListTile(
22               title: Text(pendaftaranList[index].nama),
23               subtitle: Column(
24                 crossAxisAlignment: CrossAxisAlignment.start,
25                 children: [
26                   Text(pendaftaranList[index].email),
27                   Text(pendaftaranList[index].noTelp),
28                   Text(pendaftaranList[index].jenisKelamin),
29                   Text(pendaftaranList[index].bahasa),
30                   Text(pendaftaranList[index].agama),
31                   Text(pendaftaranList[index].tanggalDaftar),
32                   Text(pendaftaranList[index].jamDaftar),
33                 ],
34               ),
35               trailing: IconButton(
36                 onPressed: () {
37                   hapusData(index);
38                 },
39                 icon: Icon(Icons.delete),
40               ),
41             ),
42           ),
43     );
44   };
45 }
46
47 void hapusData(int index) {
48   showDialog(
49     context: context,
50     builder:
51       (context) => AlertDialog(
52         title: Text("Hapus Data"),
53         content: Text(
54           "Apakah anda yakin ingin menghapus data ${pendaftaranList[index].nama}?",
55         ),
56         actions: [
57           ElevatedButton(
58             onPressed: () {
59               setState(() {
60                 pendaftaranList.removeAt(index);
61                 Navigator.pop(context);
62               });
63             },
64             child: Text("Ya"),
65           ),
66           ElevatedButton(
67             onPressed: () {
68               setState(() {
69                 Navigator.pop(context);
70               });
71             },
72             child: Text("Tidak"),
73           ),
74         ],
75       ),
76   );
77 }
78 }
79

```

Gambar 6. Kode dataPendaftaran.dart

#### 7. Main.dart

Kode utama dalam aplikasi ini yang didalamnya berisi texfield untuk nama, email, dan jenis kelamin serta kumpulan widget dari berbagai file diatas untuk menginput data sehingga dapat ditambahkan kedalam model. Dan membentuk tampilan formulir pengisian data yang komplit. dan tiap input akan divalidasi menggunakan method sudahValid dari model untuk memastikan bahwa data terisi semua dan siap di tambahkan ketika button simpan ditekan. Button simpan dan lihat data akan routing ke file dataPendaftaran untuk melihat keseluruhan data yang ada



```

1  inp
2  inp
3  inp
4  inp
5  inp
6  inp
7  inp
8
9  voi
10

```

Gambar 7. Kode main.dart