



6/4/2024

# Modul Praktikum

## Pemrograman Mobile

### **Pertemuan 11**

Pokok Bahasan :

Mengonlinekan Restful API dengan Ngrok

Mengakses RESTful API menggunakan Flutter dengan Library HTTP

### **Kebutuhan**

Komputer / Laptop

Visual Studio Code

XAMPP

Ngrok

Postman

CodeIgniter 4

Restful API yang sudah dibuat pada pertemuan sebelumnya

Browser (Google Chrome)

**Nandang Hermanto**

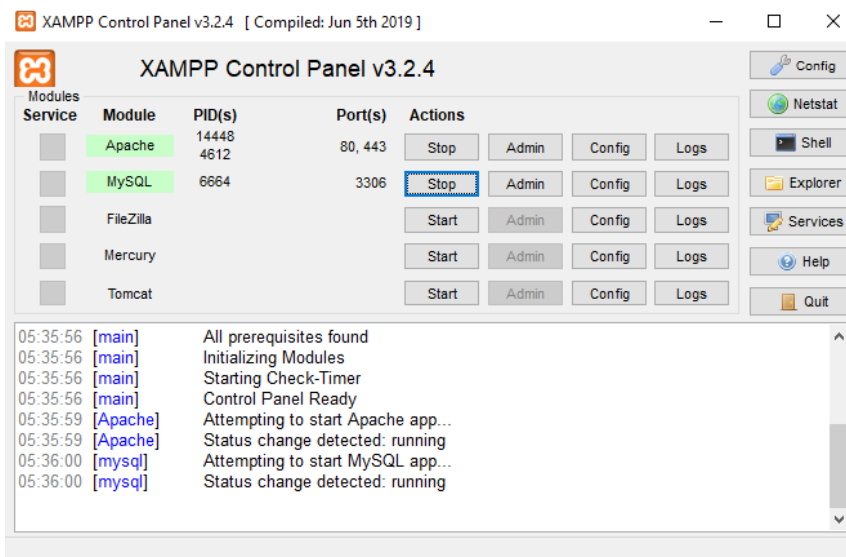
UNIVERSITAS AMIKOM PURWOKERTO

Video bisa diakses melalui link berikut ini

[https://youtu.be/Dw-4TPRN1bk?si=8owpHQ\\_bpwBlK3pB](https://youtu.be/Dw-4TPRN1bk?si=8owpHQ_bpwBlK3pB)

## Bagian 1 Mengonlinekan Restful api menggunakan ngrok

1. Pastikan XAMPP, Composer dan Postman sudah terinstall di computer anda
2. Aktifkan MySQL dan Apache



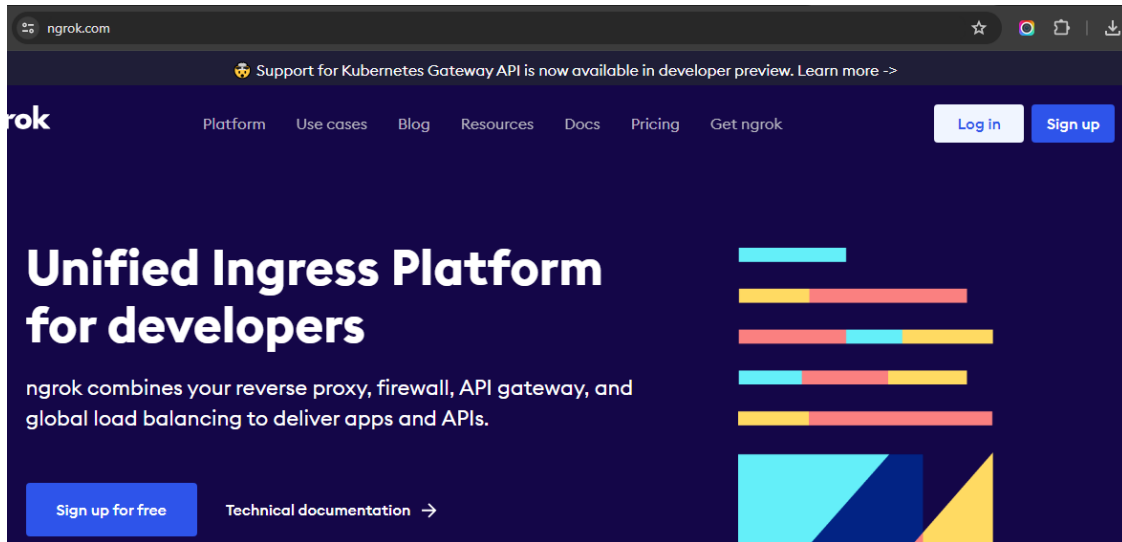
3. Jalankan resful api yang sudah dibuat pada pertemuan sebelumnya di localhost port 8080 dengan cara **php spark serve**

```
PS D:\xampp\htdocs\pmo2024api> php spark serve

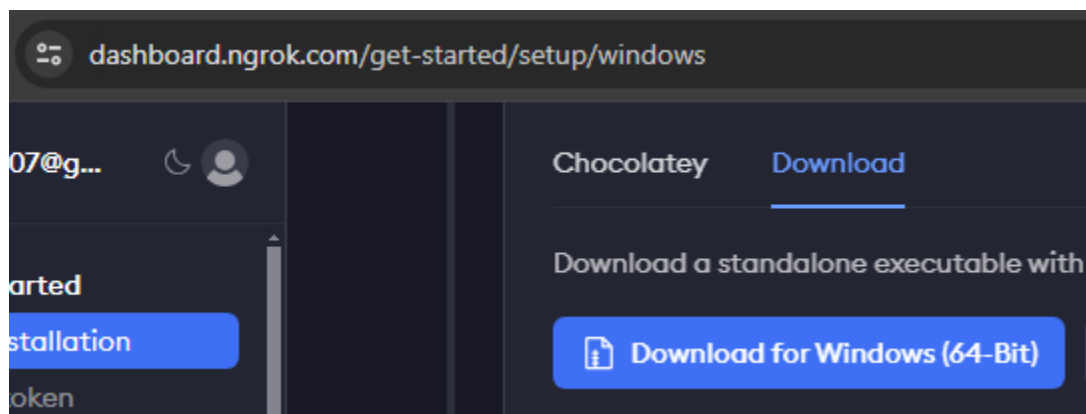
CodeIgniter v4.5.1 Command Line Tool - Server Time: 2024-06-03 22:56:56 UTC+00:00

CodeIgniter development server started on http://localhost:8080
Press Control-C to stop.
[Tue Jun  4 05:56:56 2024] PHP 8.2.12 Development Server (http://localhost:8080) started
```

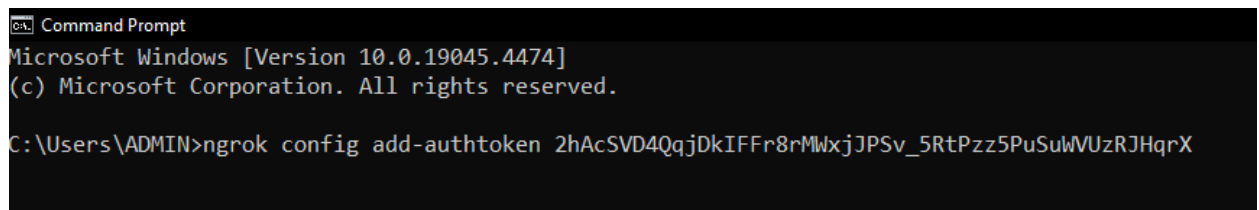
4. Langkah mengonlinekan Restful Api dengan Ngrok
  - a. Buat akun baru/daftar ke ngrok dengan mengikuti link <https://ngrok.com/>



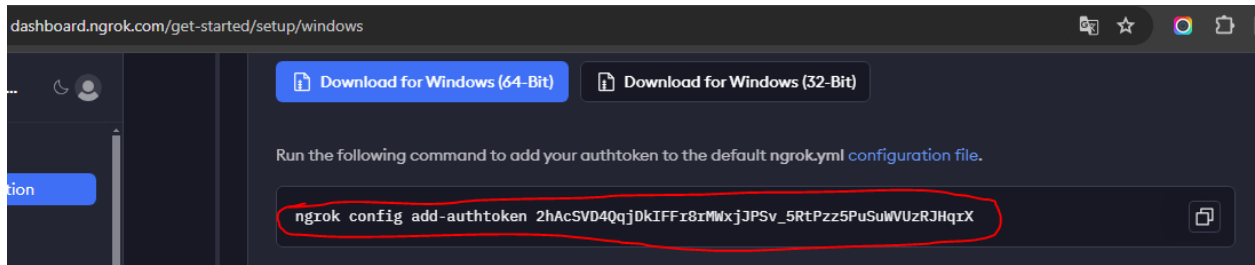
- b. Download kemudian Extract Ngrok dari link berikut ini <https://dashboard.ngrok.com/get-started/setup/windows>



- c. Tambahkan token ke **ngrok.yml** dengan menjalankan perintah berikut ini dari **command line** tempat Dimana ngrok di extract  
**ngrok config add-authtoken <<token yang anda dapatkan dari web ngrok>>**



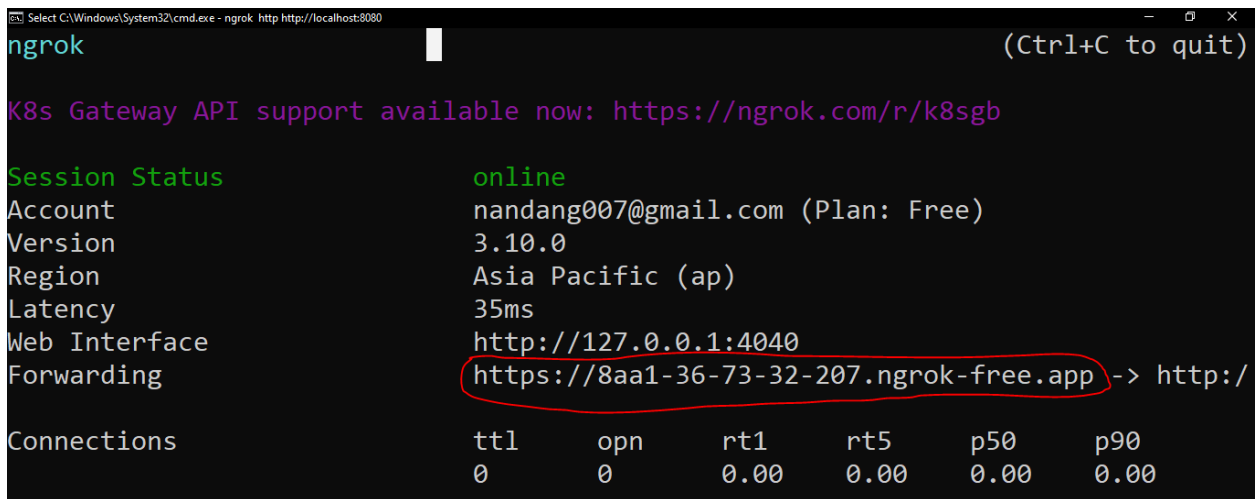
Catatan : token dapat anda peroleh dari link <https://dashboard.ngrok.com/get-started/setup/windows>



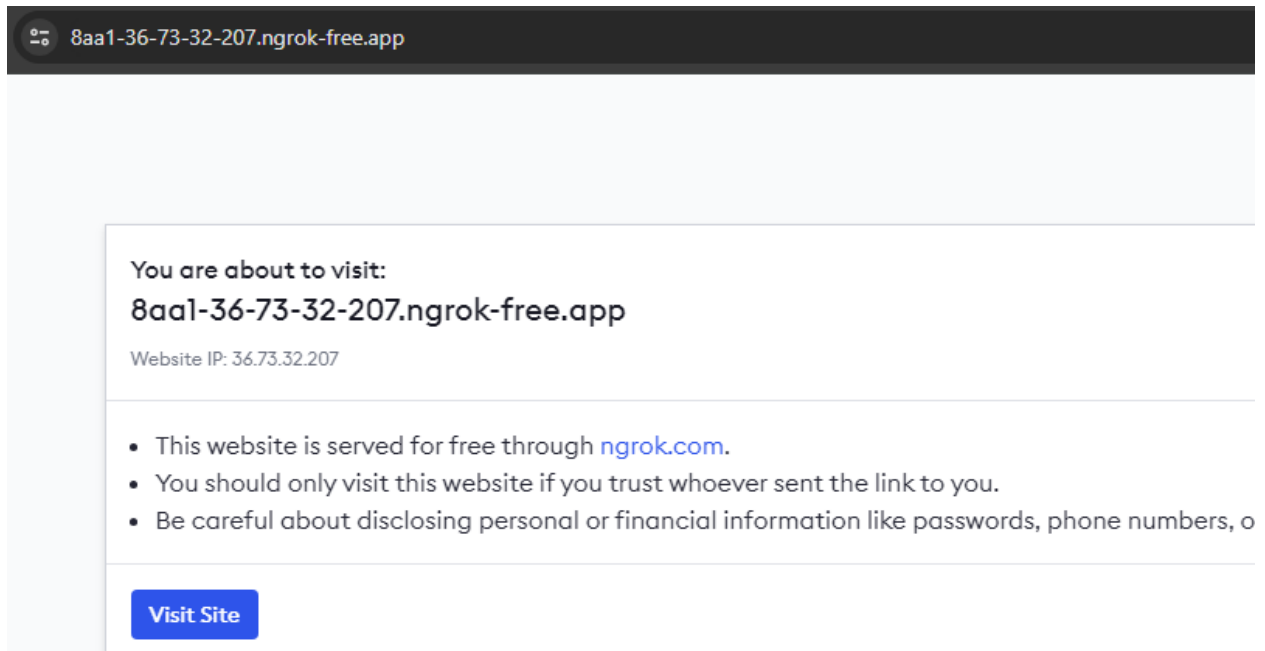
Perlu diperhatikan : token akan berbeda untuk setiap user jadi pastikan bukan menyalin dari gambar diatas

- d. Online kan restful api anda dengan mengetikan **ngrok http** <http://localhost:8080> dari command line tempat ngrok di extract

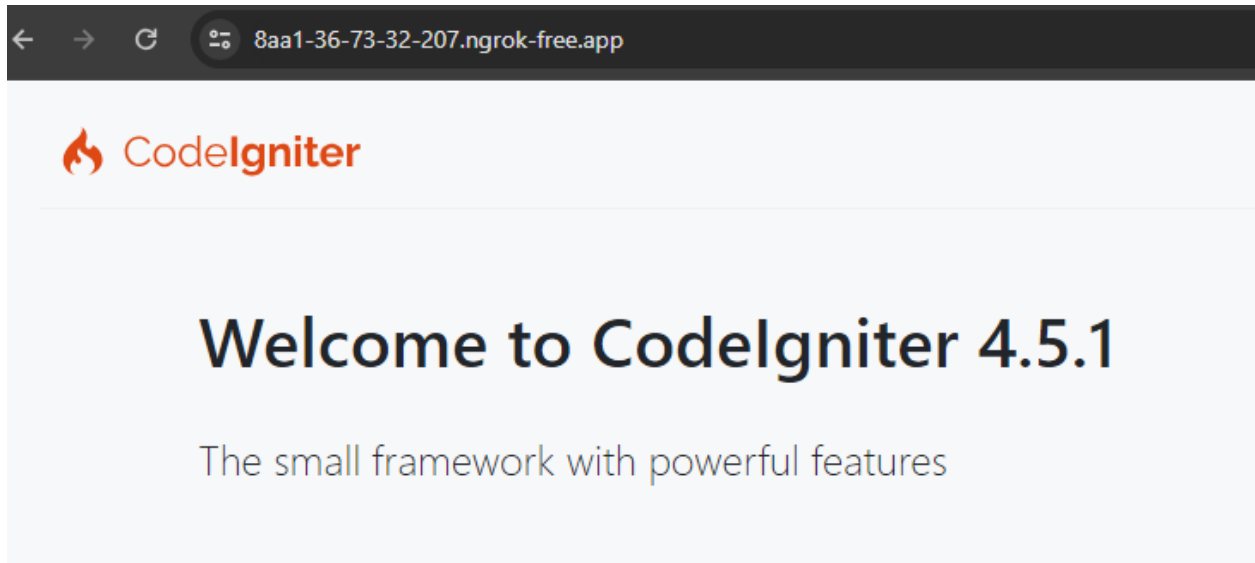
Jika sudah keluar tampilan seperti dibawah ini berarti restful api anda sudah online dan sudah bisa diakses dari jaringan internet



- e. Buka melalui browser Alamat yang diberi lingkaran merah untuk mengakses restul api anda
- f. Tekan visit site untuk menuju ke halaman restful anda



- g. Jika setelah ditekan visit site keluar halaman seperti dibawah ini bararti anda sudah berhasil mengonlinkan restful api anda



## About this page

The page you are looking at is being generated dynamically by CodeIgniter.

If you would like to edit this page you will find it located at:

```
app/Views/welcome_message.php
```

The corresponding controller for this page can be found at:

- h. Pastikan command line tempat anda menjalankan ngrok dan restful api yang dijalankan di localhost:8080 tidak ditutup selama anda mengakses restful api anda

## Bagian 2 Mengakses Restful api dari Flutter menggunakan http

1. Pastikan visual studio code dan Dart sudah siap di komputer anda
2. Jalankan Visual Studio Code
3. Buat project flutter baru dengan cara **View – Command Pallete – Flutter:New Project – Application** – arahkan ke folder tempat aplikasi akan disimpan – select a folder to create the project in – beri nama file – tekan enter
4. Buat file dart baru beri nama **konstanta.dart**

```
1  const baseUrl = "https://2d6e-36-73-32-207.ngrok-free.app/";
```

Pastikan baseUrl berisi alamat restful api anda yang benar

5. Buat file dart baru beri nama **pendaftaran\_model.dart**

```
lib\pendaftaran_model.dart

1 class PendaftaranModel {
2   String? id;
3   String? nama;
4   String? email;
5   String? noTelpon;
6   String? jenisKelamin;
7   String? bahasa;
8   String? agama;
9   String? tanggalDaftar;
10  String? jamDaftar;
11
12  PendaftaranModel(
13    {this.id,
14     this.nama,
15     this.email,
16     this.noTelpon,
17     this.jenisKelamin,
18     this.bahasa,
19     this.agama,
20     this.tanggalDaftar,
21     this.jamDaftar});
22
23  PendaftaranModel.fromJson(Map<String, dynamic> json) {
24    id = json['id'];
25    nama = json['nama'];
26    email = json['email'];
27    noTelpon = json['no_telpon'];
28    jenisKelamin = json['jenis_kelamin'];
29    bahasa = json['bahasa'];
30    agama = json['agama'];
31    tanggalDaftar = json['tanggal_daftar'];
32    jamDaftar = json['jam_daftar'];
33  }
34
35  Map<String, dynamic> toJson() {
36    final Map<String, dynamic> data = <String, dynamic>{};
37    data['id'] = id ?? "";
38    data['nama'] = nama ?? "";
39    data['email'] = email ?? "";
40    data['no_telpon'] = noTelpon ?? "";
41    data['jenis_kelamin'] = jenisKelamin ?? "";
42    data['bahasa'] = bahasa ?? "";
43    data['agama'] = agama ?? "";
44    data['tanggal_daftar'] = tanggalDaftar ?? "";
45    data['jam_daftar'] = jamDaftar ?? "";
46    return data;
47  }
48 }
49
```

## 6. Buat file dart baru beri nama **pendaftaran\_add.dart**

```
lib\pendaftaran_add.dart

1 import 'dart:convert';
2 import 'package:flutter/material.dart';
3 import 'package:flutter_http_latihan2/main.dart';
4 import 'package:flutter_http_latihan2/pendaftaran_model.dart';
5 import 'package:http/http.dart' as http;
6 import 'constant.dart';
7
8 class PendaftaranAdd extends StatefulWidget {
9   const PendaftaranAdd({super.key});
10
11   @override
12   State<PendaftaranAdd> createState() => _PendaftaranAddState();
13 }
14
15 class _PendaftaranAddState extends State<PendaftaranAdd> {
16   @override
17   Widget build(BuildContext context) {
18     return Scaffold(
19       appBar: AppBar(
20         title: Text("Tambah data Pendaftaran"),
21       ),
22       body: PopScope(
23         canPop: false,
24         onPopInvoked: (didPop) {
25           Navigator.pushReplacement(
26             context,
27             MaterialPageRoute(
28               builder: (context) => DataPendaftaran(),
29             ));
30         },
31       child: ListView(
32         children: [
33           ElevatedButton(
34             onPressed: () async {
35               var url = Uri.parse(baseUrl + 'pendaftaran');
36               var respons = await http.post(url,
37                 body: PendaftaranModel(
38                   nama: "jokoabc4",
39                   email: "jokoabc4@gmail.com",
40                   noTelpon: "08964",
41                   jenisKelamin: "Pria",
42                   bahasa: "Indonesia,Inggris,Batak",
43                   agama: "Katolik",
44                   tanggalDaftar: "2024-01-01",
45                   jamDaftar: "20:00:00",
46                 ).toJson());
47
48               Map<String, dynamic> responsDecode =
49                 json.decode(respons.body) as Map<String, dynamic>;
50
51               if (responsDecode.toString().contains('error')) {
52                 ScaffoldMessenger.of(context).showSnackBar(SnackBar(
53                   content: Text(responsDecode['messages']
54                     .toString()
55                     .replaceAll(", ", "\n\n"))));
56               } else {
57                 Navigator.pushReplacement(
```

```
58         context,
59         MaterialPageRoute(
60           builder: (context) => DataPendaftaran(),
61         ));
62       }
63
64       // print(responsDecode);
65     },
66     child: Text("Simpan"),
67   ),
68 ],
69 ),
70 ),
71 );
72 }
73 }
74 }
```

7. Buka File **main.dart** kemudian edit menjadi seperti dibawah ini

lib\main.dart

```

1  import 'dart:convert';
2
3  import 'package:flutter/material.dart';
4  import 'package:flutter_http_latihan2/constant.dart';
5  import 'package:flutter_http_latihan2/pendaftaran_add.dart';
6  import 'package:flutter_http_latihan2/pendaftaran_model.dart';
7  import 'package:http/http.dart' as http;
8
9  void main(List<String> args) {
10     runApp(MyApp());
11 }
12
13 class MyApp extends StatelessWidget {
14     const MyApp({super.key});
15
16     @override
17     Widget build(BuildContext context) {
18         return MaterialApp(
19             debugShowCheckedModeBanner: false,
20             home: DataPendaftaran(),
21         );
22     }
23 }
24
25 class DataPendaftaran extends StatefulWidget {
26     const DataPendaftaran({super.key});
27
28     @override
29     State<DataPendaftaran> createState() => _DataPendaftaranState();
30 }
31
32 class _DataPendaftaranState extends State<DataPendaftaran> {
33     List<PendaftaranModel> pendafatarnList = [];
34
35     Future ambilDataPendaftaran() async {
36         // await Future.delayed(Duration(seconds: 5));
37         var url = Uri.parse(baseUrl + 'pendaftaran');
38         var respons = await http.get(url);
39         List responsDecode = json.decode(respons.body);
40         pendafatarnList.clear();
41         for (var element in responsDecode) {
42             pendafatarnList.add(PendaftaranModel.fromJson(element));
43         }
44         // print(responsDecode);
45         // print(pendafatarnList.length);
46         // print(pendafatarnList[0].nama);
47     }
48
49     @override
50     Widget build(BuildContext context) {
51         return Scaffold(
52             appBar: AppBar(
53                 title: Text("Data pendaftaran"),
54             ),
55             body: FutureBuilder(
56                 future: ambilDataPendaftaran(),
57                 builder: (context, snapshot) {

```

```

58     if (snapshot.connectionState == ConnectionState.waiting) {
59         return Center(
60             child: CircularProgressIndicator(),
61         );
62     } else {
63         if (pendaftarnList.length == 0) {
64             return Center(
65                 child: Text("Data tidak ditemukan"),
66             );
67         }
68
69         return ListView.builder(
70             itemCount: pendaftarnList.length,
71             itemBuilder: (context, index) => Card(
72                 color: Colors.green.shade50,
73                 child: ListTile(
74                     trailing: Container(
75                         width: 100,
76                         child: Row(
77                             children: [
78                                 IconButton(
79                                     onPressed: () async {
80                                         showDialog(
81                                             context: context,
82                                             builder: (context) => AlertDialog(
83                                                 title: Text("Hapus Data"),
84                                                 content: Text(
85                                                     "Apakah anda yakin akan menghapus data"),
86                                                 actions: [
87                                                     ElevatedButton(
88                                                         onPressed: () async {
89                                                             var url = Uri.parse(baseUrl +
90                                                                 'pendaftaran/${pendaftarnList[index].id!}');
91                                                             var respons = await http.delete(url);
92                                                             setState(() {});
93                                                             Navigator.pop(context);
94                                                         },
95                                                         child: Text("Ya")),
96                                                     ElevatedButton(
97                                                         onPressed: () {
98                                                             Navigator.pop(context);
99                                                         },
100                                                         child: Text("Tidak")),
101                                                 ],
102                                             ),
103                                         );
104                                     },
105                                     icon: Icon(Icons.delete),
106                                 ),
107                                 IconButton(
108                                     onPressed: () async {
109                                         showDialog(
110                                             context: context,
111                                             builder: (context) => AlertDialog(
112                                                 title: Text("Edit Data"),
113                                                 content: Text(
114                                                     "Apakah anda yakin akan Mengedit data"),
115                                                 actions: [
116                                                     ElevatedButton(
117                                                         onPressed: () async {

```

```

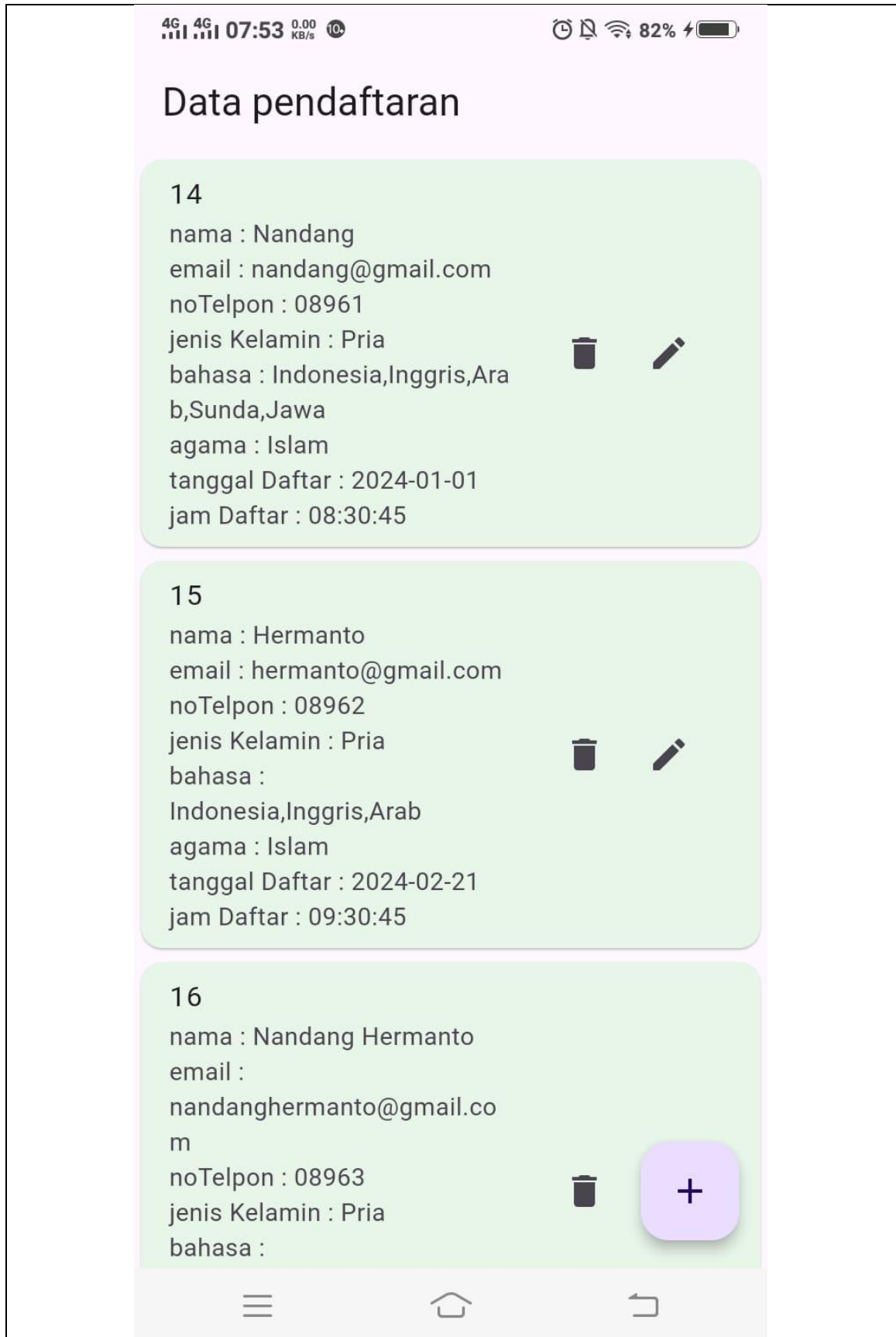
118         var url = Uri.parse(baseUrl +
119             'pendaftaran/${pendaftarnList[index].id!}');
120
121         var respons = await http.put(url,
122             body: PendaftaranModel(
123                 nama: "Joko Edit",
124                 email: "Jokoedit@gmail.com",
125                 noTelpon: "08964",
126                 jenisKelamin: "Pria",
127                 bahasa: "Indonesia,Inggris,Batak",
128                 agama: "Katolik",
129                 tanggalDaftar: "2024-01-01",
130                 jamDaftar: "20:00:00",
131             ).toJson());
132         setState(() {});
133         Navigator.pop(context);
134     },
135     child: Text("Ya")),
136     ElevatedButton(
137         onPressed: () {
138             Navigator.pop(context);
139         },
140         child: Text("Tidak")),
141     ],
142 ),
143 );
144 },
145 icon: Icon(Icons.edit),
146 ),
147 ],
148 ),
149 ),
150 title: Text(pendaftarnList[index].id!),
151 subtitle: Column(
152     crossAxisAlignment: CrossAxisAlignment.start,
153     children: [
154         Text('nama : ${pendaftarnList[index].nama!}'),
155         Text('email : ${pendaftarnList[index].email!}'),
156         Text('noTelpon : ${pendaftarnList[index].noTelpon!}'),
157         Text(
158             'jenis Kelamin : ${pendaftarnList[index].jenisKelamin!}'),
159         Text('bahasa : ${pendaftarnList[index].bahasa!}'),
160         Text('agama : ${pendaftarnList[index].agama!}'),
161         Text(
162             'tanggal Daftar : ${pendaftarnList[index].tanggalDaftar!}'),
163         Text('jam Daftar : ${pendaftarnList[index].jamDaftar!}'),
164     ],
165 ),
166 ),
167 ),
168 );
169 }
170 },
171 ),
172 floatingActionButton: FloatingActionButton(
173     onPressed: () {
174         Navigator.pushReplacement(
175             context,
176             MaterialPageRoute(
177                 builder: (context) => PendaftaranAdd(),

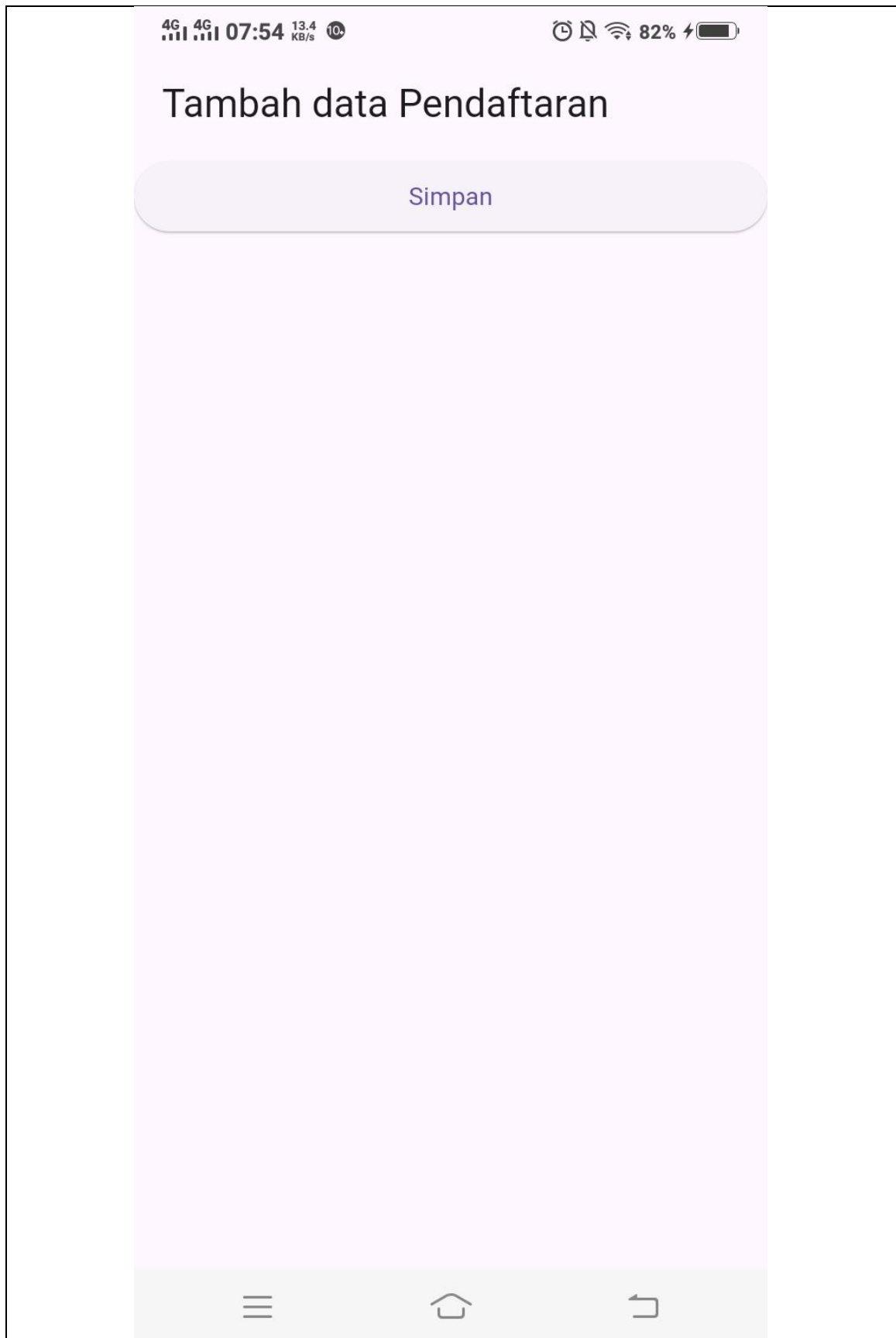
```

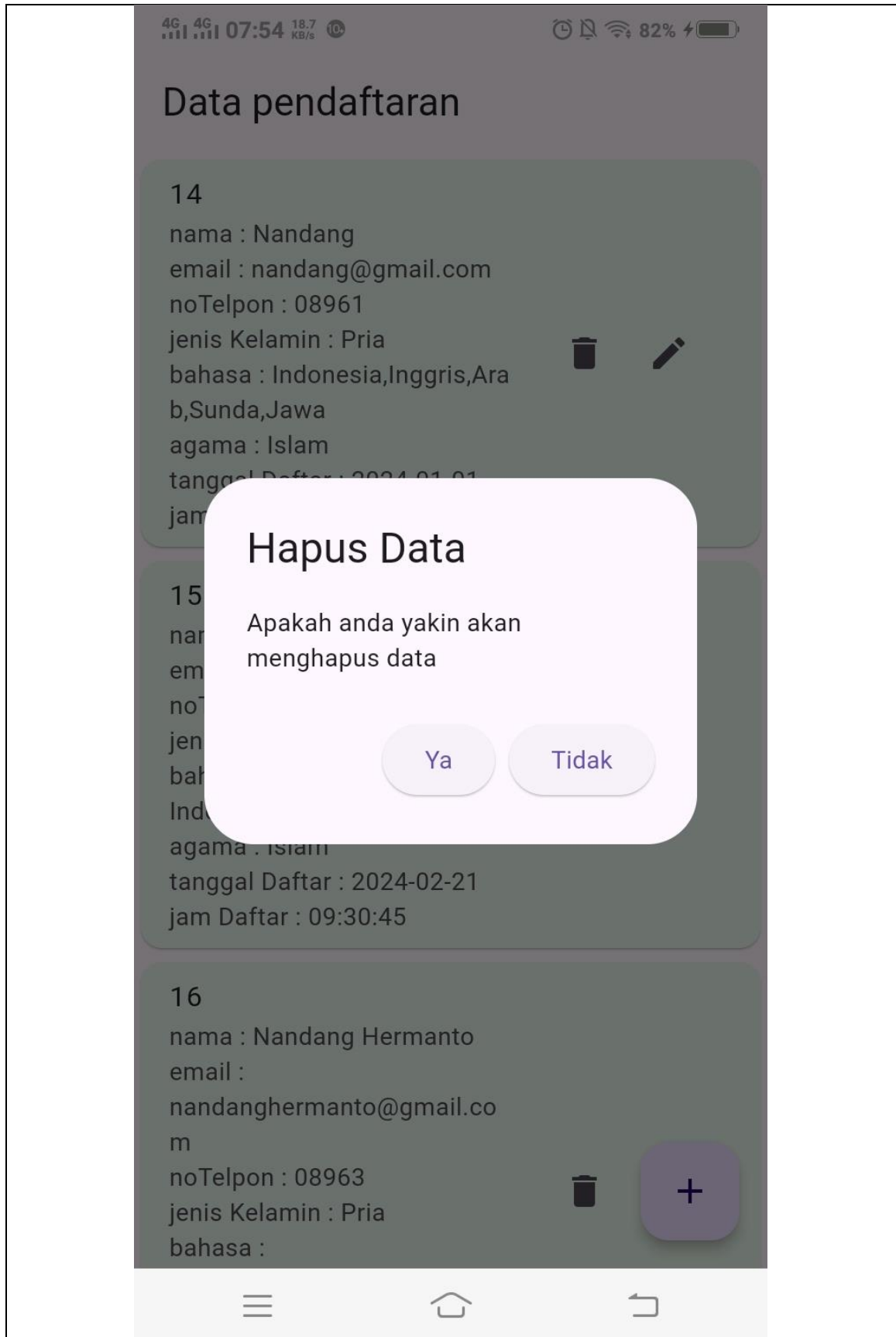
```
178         ));  
179     },  
180     child: Icon(Icons.add),  
181   ),  
182 );  
183 }  
184 }  
185
```

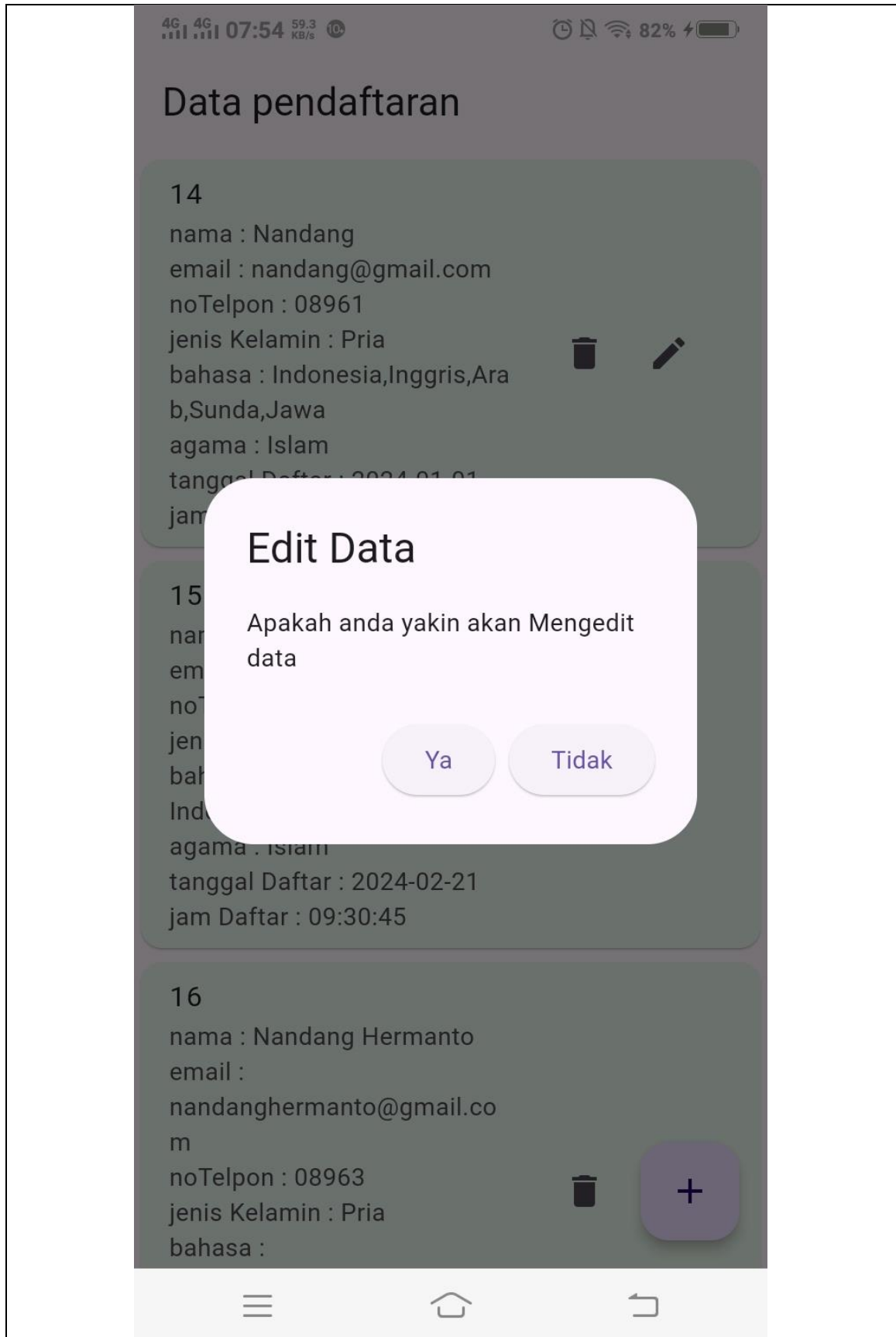
8. Jalankan program

Hasil Program



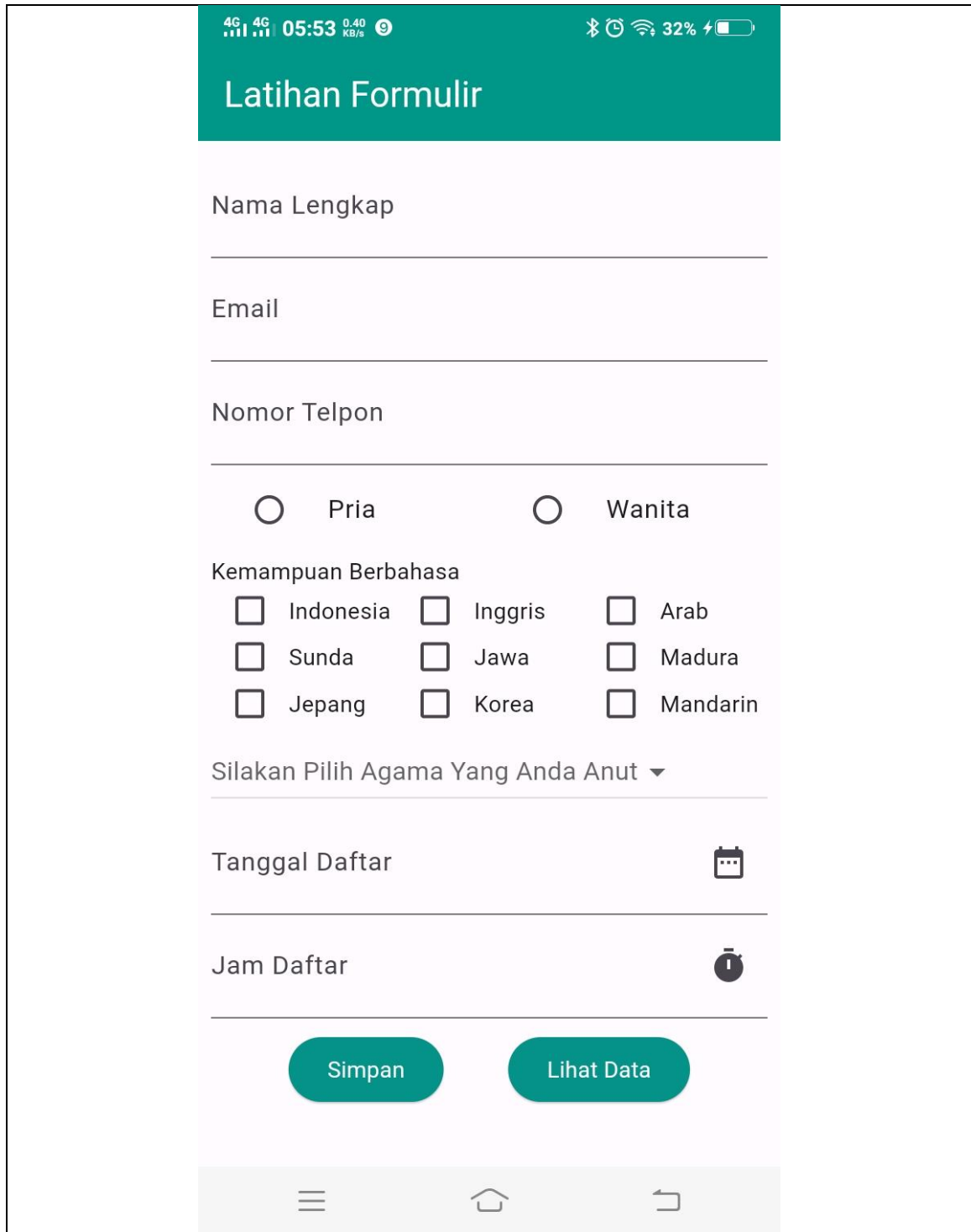






## 9. Tantangan

Perbaiki form tambah data dan edit data menggunakan materi formulir sehingga tampilannya menjadi seperti berikut ini



The screenshot shows a mobile application interface with a teal header bar containing the title "Latihan Formulir". Below the header, the form consists of several input fields and sections:

- Nama Lengkap**: A text input field.
- Email**: A text input field.
- Nomor Telpon**: A text input field.
- Gender**: Two radio button options, "Pria" and "Wanita".
- Kemampuan Berbahasa**: A section with a grid of checkboxes for various languages: Indonesia, Inggris, Arab, Sunda, Jawa, Madura, Jepang, Korea, and Mandarin.
- Religion**: A dropdown menu labeled "Silakan Pilih Agama Yang Anda Anut".
- Tanggal Daftar**: A date input field with a calendar icon.
- Jam Daftar**: A time input field with a clock icon.
- Buttons**: Two teal buttons at the bottom, "Simpan" (Save) and "Lihat Data" (View Data).

The bottom of the screen features a standard Android navigation bar with icons for a menu, home, and back.