

RESPONSI 1 PEMROGRAMAN MOBILE
PANKY BINTANG PRADANA YOSUA
H1D022077 | SHIFT LAMA D | SHIFT BARU F

H1D022077

Hasil Modulo NIM : 1

3 digit terakhir : 077

Segment Satu

1. Paket: 1
2. Jenis Aplikasi: Aplikasi Manajemen Kesehatan

Segment Dua

1. Digit Puluhan: 7
2. Nama Tabel: riwayat_alergi
3. Kolom 1: id (int, PK, increment)
4. Kolom 2: allergen (String)
5. Kolom 3: reaction (String)
6. Kolom 4: severity_scale (Integer)

Segment Tiga

1. Digit Satuan: 7
2. Kustomisasi Tampilan UI: Tema Warna Pastel, Font Calibri

1. Registrasi lib\ui\registrasi_page.dart

Halaman ini digunakan untuk mendaftarkan user

a. Halaman Registrasi

Halaman registrasi dirender oleh kode di bawah.

```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      title: const Text("Registrasi"),  
    ),  
    body: SingleChildScrollView(  
      child: Padding(  
        padding: const EdgeInsets.all(8.0),  
        child: Form(  
          key: _formKey,  
          child: Column(  
            mainAxisAlignment: MainAxisAlignment.center,  
            children: [  
              Text(  
                'Silahkan registrasi terlebih dahulu untuk mendaftarkan  
diri ke Aplikasi Manajemen Riwayat Alergi',  
                style: Theme.of(context).textTheme.bodyMedium),  
              _namaTextField(),  
              _emailTextField(),  
              _passwordTextField(),  
              _passwordKonfirmasiTextField(),  
            ],  
          ),  
        ),  
      ),  
    ),  
  );  
}
```

```

        const SizedBox(height: 10),
        _buttonRegistrasi()
      ],
    ),
  ),
),
),
);
}

```

2:17

← Registrasi

Silahkan registrasi terlebih dahulu untuk mendaftarkan diri ke Aplikasi Manajemen Riwayat Alergi

Nama
Katniss

Email
katniss@bintangyosua.my.id

Password
.....

Konfirmasi Password
.....

Registrasi

1 2 3 4 5 6 7 8 9 0
q w e r t y u i o p
a s d f g h j k l
↑ z x c v b n m ⊞
?123 , . ✓

b. Registrasi Sukses

Registrasi dilakukan dengan klik tombol Registrasi setelah mengisi data. Proses yang dilakukan ada pada kode di bawah.

Set loading menjadi true, lalu data pada registrasi ditangkap, lalu memanggil fungsi `registrasi()` pada `RegistrasiBloc`. Pada proses ini dilakukan request api POST ke path `/api/registrasi` dengan data yang dikirimkan yaitu nama, email, dan password.

Ada dua kondisi yang dapat terjadi, jika sukses akan memunculkan pesan Registrasi berhasil, silahkan login sedangkan jika gagal akan memunculkan pesan Registrasi gagal, silahkan coba lagi.

```
void _submit() {
  _formKey.currentState!.save();
  setState(() {
    _isLoading = true;
  });
  RegistrasiBloc.registrasi(
    nama: _namaTextboxController.text,
    email: _emailTextboxController.text,
    password: _passwordTextboxController.text)
    .then((value) {
      showDialog(
        context: context,
        barrierDismissible: false,
        builder: (BuildContext context) => SuccessDialog(
          description: "Registrasi berhasil, silahkan login",
          okClick: () {
            Navigator.pop(context);
          },
        ),
      );
    }, onError: (error) {
      showDialog(
        context: context,
        barrierDismissible: false,
        builder: (BuildContext context) => const WarningDialog(
          description: "Registrasi gagal, silahkan coba lagi",
        ),
      );
    });
}
```

2:18



Registrasi

Silahkan registrasi terlebih dahulu untuk mendaftarkan diri ke Aplikasi Manajemen Riwayat Alergi

Nama

Katniss

Email

katniss@bintangyosua.my.id

Password

•••••

Konfirmasi Password

•••••

Registrasi

SUKSES

Registrasi berhasil, silahkan login

OK

2. Login `lib\ui\login_page.dart`

Halaman login digunakan untuk masuk ke aplikasi, sehingga dapat melakukan proses CRUD pada aplikasi Manajemen Riwayat Alergi.

a. Halaman Login

Kode yang dirender adalah sebagai berikut.

```
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('Login'),
    ),
    body: SingleChildScrollView(
      child: Padding(
        padding: const EdgeInsets.all(8.0),
        child: Form(
          key: _formKey,
          child: Column(
            children: [
              Text(
                'Silahkan login terlebih dahulu untuk masuk ke Aplikasi
                Manajemen Riwayat Alergi',
                style: Theme.of(context).textTheme.bodyMedium),
              _emailTextField(),
              _passwordTextField(),
              const SizedBox(
                height: 10,
              ),
              _buttonLogin(),
              const SizedBox(
                height: 10,
              ),
              _menuRegistrasi()
            ],
          ),
        ),
      ),
    ),
  );
}
```

2:30



Login

Silahkan login terlebih dahulu untuk masuk ke Aplikasi Manajemen Riwayat Alergi

Email

katniss@bintangyosua.my.id

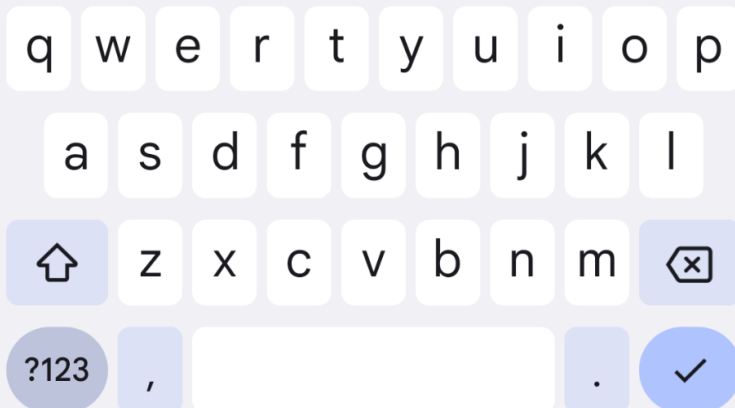
Password

.....

Login

[Registrasi](#)

Widget build(Buil...



b. Login Gagal

Proses login dilakukan dengan klik tombol **Login**. Kode yang memproses hal ini adalah sebagai di bawah.

Set loading menjadi **true**, lalu proses login melakukan request api **POST** ke path **/api/login** dengan mengirimkan email dan password. Jika status berupa **200**, proses login akan menyimpan token dan ID user yang diberikan oleh server hasil dari request API, lalu diarahkan ke halaman list riwayat alergi. Jika kode selain **200** maka akan dimunculkan pop up pesan **Login gagal, silahkan coba lagi**. Namun, jika terjadi error pada proses login akan memunculkan pesan yang sama juga.

```
void _submit() {
  _formKey.currentState!.save();
  setState(() {
    _isLoading = true;
  });

  LoginBloc.login(
    email: _emailTextboxController.text,
    password: _passwordTextboxController.text)
    .then((value) async {
      if (value.code == 200) {
        await UserInfo().setToken(value.token.toString());
        await UserInfo().setUserID(int.parse(value.userID.toString()));
        Navigator.pushReplacement(context,
          MaterialPageRoute(builder: (context) => const AlergiPage()));
      } else {
        showDialog(
          context: context,
          barrierDismissible: false,
          builder: (BuildContext context) => const WarningDialog(
            description: "Login gagal, silahkan coba lagi",
          ));
      }
    }, onError: (error) {
      showDialog(
        context: context,
        barrierDismissible: false,
        builder: (BuildContext context) => const WarningDialog(
          description: "Login gagal, silahkan coba lagi",
        ));
    });
}
```

2:33



Login

Silahkan login terlebih dahulu untuk masuk ke Aplikasi Manajemen Riwayat Alergi

Email

katniss@bintangyosua.my.id

Password

•••••

Login

[Registrasi](#)

GAGAL

Login gagal, silahkan coba lagi

OK

3. List Riwayat Alergi `lib\ui\alergi_page.dart`

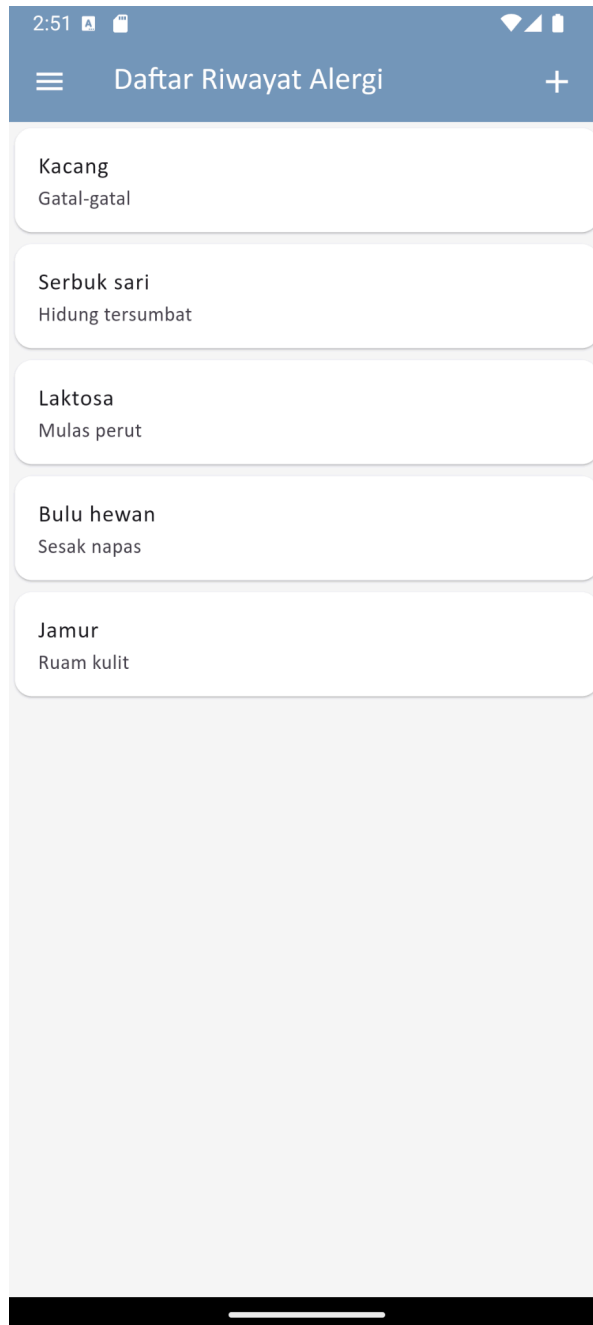
Sebelum merender halaman list riwayat alergi, akan dicek terlebih dahulu apakah ada token pada storage. Jika ada akan diarahkan ke halaman list riwayat alergi. Sedangkan jika tidak ada token maka diarahkan pada halaman login untuk melakukan proses login supaya mendapatkan token, sehingga dapat mengunjungi halaman list riwayat alergi. Lalu, mengambil data pada path `/api/kesehatan/riwayat_alergi`

```
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('Daftar Riwayat Alergi'),
      actions: [
        Padding(
          padding: const EdgeInsets.only(right: 20.0),
          child: GestureDetector(
            child: const Icon(Icons.add, size: 26.0),
            onTap: () async {
              Navigator.push(context,
                MaterialPageRoute(builder: (context) => AlergiForm()));
            },
          ),
        ),
      ],
    ),
    drawer: Drawer(
      backgroundColor: const Color.fromRGB(165, 182, 141, 1),
      child: ListView(
        children: [
          ListTile(
            title: const Text('Logout'),
            trailing: const Icon(Icons.logout),
            onTap: () async {
              await LogoutBloc.logout().then((value) => {
                Navigator.of(context).pushAndRemoveUntil(
                  MaterialPageRoute(builder: (context) =>
LoginPage()),
                    (route) => false);
              });
            },
          ),
        ],
      ),
    ),
    body: FutureBuilder<List>(
      future: AlergiBloc.getAlergis(),
      builder: (context, snapshot) {
        if (snapshot.hasError) print(snapshot.error);
        return snapshot.hasData
          ? ListAlergi(
```

```

        list: snapshot.data,
      )
    : const Center(
      child: CircularProgressIndicator(),
    );
  },
),
);
}

```



4. Tambah Riwayat Alergi `lib\ui\alergi_form.dart`

Halaman ini digunakan untuk menambah riwayat alergi baru.

a. Halaman Tambah Riwayat Alergi

Halaman tambah riwayat alergi dirender pada kode di bawah.

```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(title: Text(judul)),  
    body: SingleChildScrollView(  
      child: Padding(  
        padding: const EdgeInsets.all(8.0),  
        child: Form(  
          key: _formKey,  
          child: Column(  
            children: [  
              _kodeAlergiTextField(),  
              _namaAlergiTextField(),  
              _hargaAlergiTextField(),  
              const SizedBox(height: 10),  
              _buttonSubmit()  
            ],  
          ),  
        ),  
      ),  
    ),  
  );  
};  
}
```

2:43



Tambah Riwayat Alergi

Kode Alergi

Peanuts

Nama Alergi

Rash

Harga

3

Simpan



b. Tambah Riwayat Alergi Sukses

Proses menambahkan riwayat alergi dilakukan pada kode di bawah,

Pertama, set loading menjadi `true`. Lalu membuat objek riwayat alergi dengan atribut `allergen`, `reaction`, dan `severity_scale`. Setelah itu request api `POST` ke path `/api/kesehatan/riwayat_alergi` dengan data yang dikirimkan dari atribut object Alergi. Jika sukses akan diarahkan ke halaman list riwayat alergi. Jika gagal akan memunculkan pesan `Simpan gagal, silahkan coba lagi`.

```
simpan() {
  setState(() {
    _isLoading = true;
  });
  Alergi createAlergi = Alergi(id: null);
  createAlergi.allergen = _kodeAlergiTextboxController.text;
  createAlergi.reaction = _namaAlergiTextboxController.text;
  createAlergi.severity_scale =
int.parse(_hargaAlergiTextboxController.text);
  AlergiBloc.addAlergi(alergi: createAlergi).then((value) {
    Navigator.of(context).push(MaterialPageRoute(
      builder: (BuildContext context) => const AlergiPage()));
  }, onError: (error) {
    showDialog(
      context: context,
      builder: (BuildContext context) => const WarningDialog(
        description: "Simpan gagal, silahkan coba lagi",
      ));
  });
  setState(() {
    _isLoading = false;
  });
}
```

2:44



Daftar Riwayat Alergi



Kacang
Gatal-gatal

Serbuk sari
Hidung tersumbat

Laktosa
Mulas perut

Bulu hewan
Sesak napas

Jamur
Ruam kulit

Peanuts
Rash

5. Detail Riwayat Alergi `lib\ui\alergi_detail.dart`

Halaman ini untuk menunjukkan detail riwayat alergi.

Di bawah ini merupakan kode untuk merender halaman detail riwayat alergi. Detail riwayat alergi memperlihatkan allergen, reaction, dan severity scale, serta tombol edit dan delete.

```
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('Detail Riwayat Alergi'),
    ),
    body: Center(
      child: Column(
        children: [
          ListTile(
            leading: const Icon(Icons.local_hospital),
            title:
              Text('Allergen: ${widget.alergi.allergen}' ?? 'Tanpa
nama'),
            subtitle: const Text('Nama alergen yang menyebabkan reaksi'),
          ),
          ListTile(
            leading: const Icon(Icons.sick),
            title:
              Text('Reaction: ${widget.alergi.reaction}' ?? 'Tanpa
nama'),
            subtitle: const Text(
              'Jenis reaksi yang terjadi akibat kontak dengan
alergen.'),
          ),
          ListTile(
            leading: const Icon(Icons.numbers),
            title: Text(
              'Severity Scale:
${widget.alergi.severity_scale.toString()}'),
            subtitle: const Text(
              'Tingkat keparahan reaksi, biasanya dalam skala 1 hingga
5, di mana 1 adalah ringan dan 5 adalah sangat serius.'),
          ),
          const SizedBox(
            height: 10,
          ),
          _tombolHapusEdit()
        ],
      ),
    ),
  );
}
```

2:54



Detail Riwayat Alergi



Allergen: Peanuts

Nama alergen yang menyebabkan reaksi



Reaction: Rash

Jenis reaksi yang terjadi akibat kontak dengan alergen.



Severity Scale: 3

Tingkat keparahan reaksi, biasanya dalam skala 1 hingga 5, di mana 1 adalah ringan dan 5 adalah sangat serius.

Edit

Delete

6. Edit Riwayat Alergi `lib\ui\alergi_form.dart`

Halaman edit riwayat alergi juga menggunakan alergi form, sama seperti saat menambahkan riwayat alergi. Namun, form pada edit riwayat alergi memiliki nilai awal. Nilai ini didapatkan dari object riwayat alergi yang dikirim dari detail riwayat alergi.`

Kode untuk memproses edit adalah sebagai berikut.

State loading set menjadi `true`. Menyimpan object riwayat alergi dengan data yang akan diubah adalah allergen, reaction, dan severity_scale berdasarkan id riwayat alergi yang dimaksud. Lalu request API PUT ke path `/riwayat_alergi/{id}/update`. Jika berhasil akan diarahkan ke halaman list riwayat alergi. Jika gagal akan dimunculkan pesan `Permintaan ubah data gagal, silahkan coba lagi`




```
ubah() {
  setState(() {
    _isLoading = true;
  });
  Alergi updateAlergi = Alergi(id: widget.alergi!.id!);
  updateAlergi.allergen = _kodeAlergiTextboxController.text;
  updateAlergi.reaction = _namaAlergiTextboxController.text;
  updateAlergi.severity_scale =
int.parse(_hargaAlergiTextboxController.text);
  AlergiBloc.updateAlergi(alergi: updateAlergi).then((value) {
    Navigator.of(context).push(MaterialPageRoute(
      builder: (BuildContext context) => const AlergiPage()));
  }, onError: (error) {
    showDialog(
      context: context,
      builder: (BuildContext context) => const WarningDialog(
        description: "Permintaan ubah data gagal, silahkan coba lagi",
      ));
  });
  setState(() {
    _isLoading = false;
  });
}
```

Kode Alergi
Peanutsss

Nama Alergi
Rashhh

Harga
5

Ubah

-  **Allergen: Peanutsss**
Nama alergen yang menyebabkan reaksi
-  **Reaction: Rashhh**
Jenis reaksi yang terjadi akibat kontak dengan alergen.
-  **Severity Scale: 5**
Tingkat keparahan reaksi, biasanya dalam skala 1 hingga 5, di mana 1 adalah ringan dan 5 adalah sangat serius.

Edit

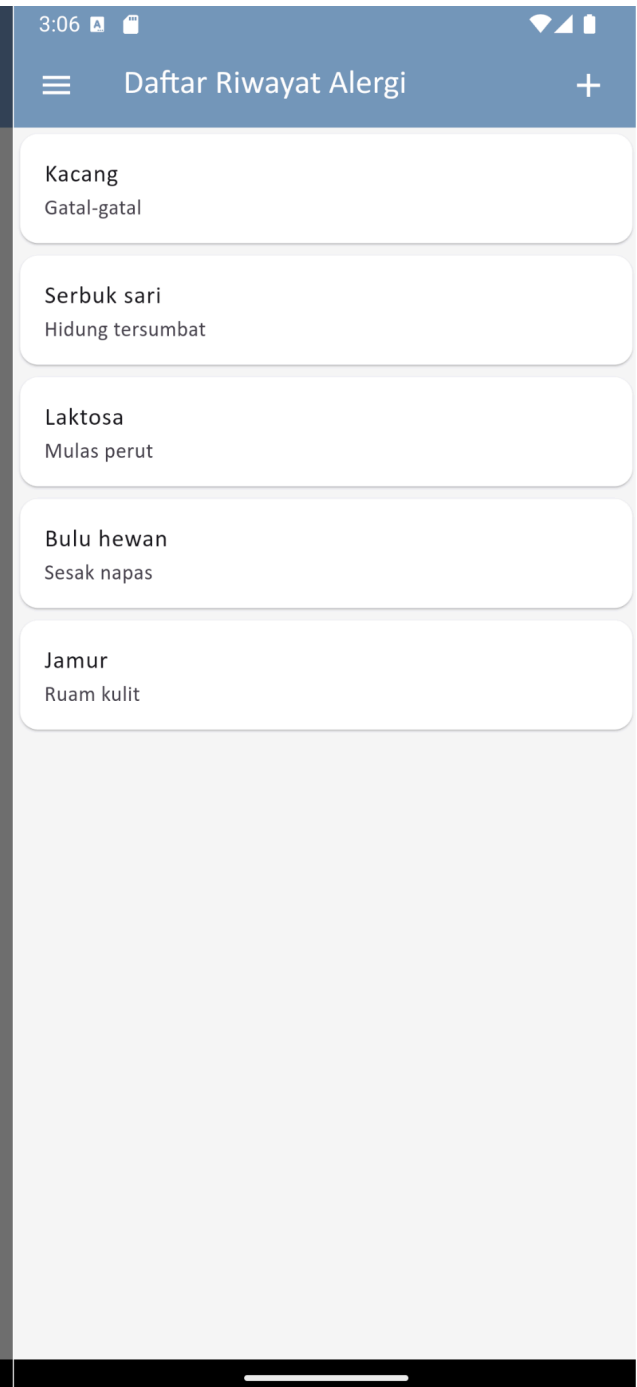
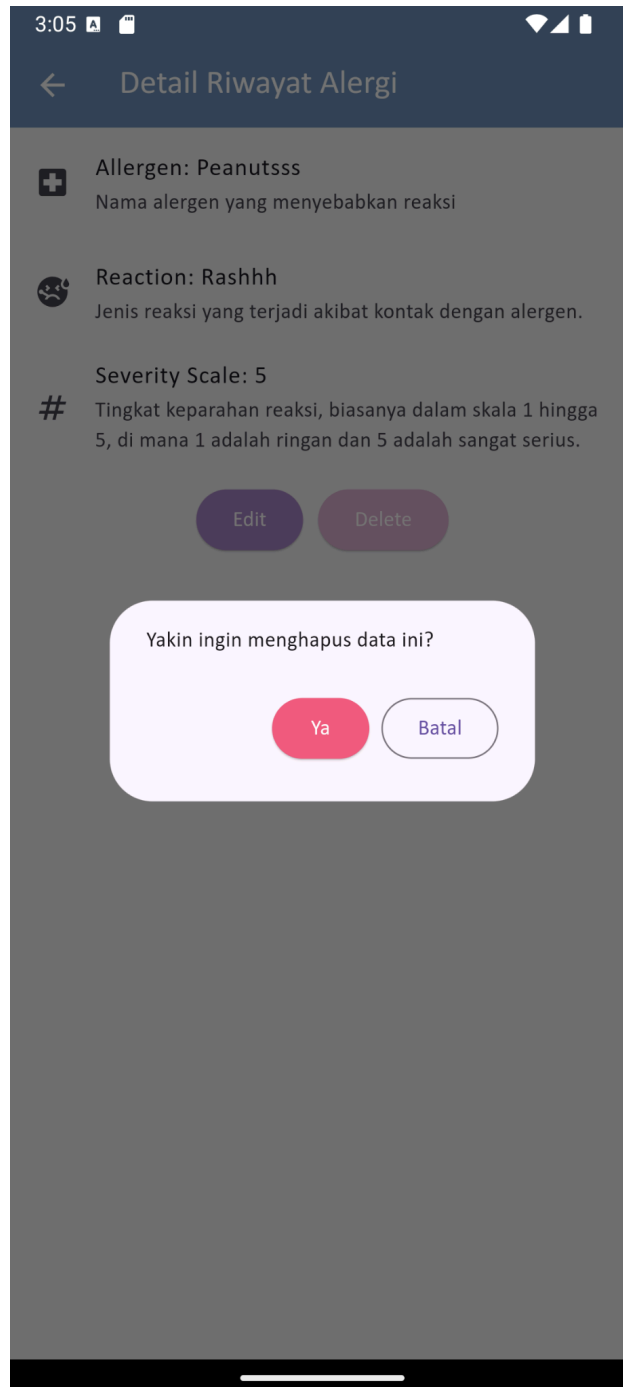
Delete

7. Pop Up Hapus Riwayat Alergi

Pop up hapus riwayat alergi dilakukan ketika meng-klik tombol hapus pada halaman detail riwayat alergi. Kode yang digunakan sebagai berikut:

```
void confirmHapus() {
  AlertDialog alertDialog = AlertDialog(
    content: const Text("Yakin ingin menghapus data ini?"),
    actions: [
      //tombol hapus
      ElevatedButton(
        style: ButtonStyle(
          backgroundColor:
            WidgetStateProperty.all(const Color(0xFFFF05A7E))),
        child: const Text(
          "Ya",
          style: TextStyle(color: Colors.white),
        ),
        onPressed: () {
          AlergiBloc.deleteAlergi(id: widget.alergi.id!).then(
            (value) => {
              Navigator.of(context).push(MaterialPageRoute(
                builder: (context) => const AlergiPage()))
            }, onError: (error) {
              showDialog(
                context: context,
                builder: (BuildContext context) => const WarningDialog(
                  description: "Hapus gagal, silahkan coba lagi",
                ));
            });
        },
      ),
      //tombol batal
      OutlinedButton(
        child: const Text("Batal"),
        onPressed: () => Navigator.pop(context),
      ),
    ],
  );
  showDialog(builder: (context) => alertDialog, context: context);
}
```

Proses ini dilakukan dengan meng-klik tombol **Ya** sebagai konfirmasi. Setelah itu menjalankan fungsi **deleteAlergi** yang mana akan request API **DELETE** ke path **/api/kesehatan/riwayat_alergi/{id}/delete**. Jika berhasil akan diarahkan ke halaman list riwayat alergi. Jika gagal akan dimunculkan pesan gagal berupa **Hapus gagal, silahkan coba lagi**. Jika meng-klik tombol **Batal** akan membatalkan konfirmasi **Hapus** riwayat alergi.



8. Sidebar dan Logout

Sidebar digunakan sebagai bar pada sisi kiri yang akan muncul dari kiri apabila hamburger icon diklik. Pada sidebar akan terdapat pilihan logout. Logout digunakan untuk mengakhiri sesi user.

Komponen sidebar dan login dirender pada kode berikut.

```
drawer: Drawer(  
  backgroundColor: const Color.fromRGBO(165, 182, 141, 1),  
  child: ListView(  
    children: [  
      ListTile(  
        title: const Text('Logout'),  
        trailing: const Icon(Icons.logout),  
        onTap: () async {  
          await LogoutBloc.logout().then((value) => {  
            Navigator.of(context).pushAndRemoveUntil(  
              MaterialPageRoute(builder: (context) => LoginPage()),  
              (route) => false)  
            });  
          },  
        ),  
      ],  
    ),  
  ),  
,
```

Proses logout dilakukan dengan meng-klik tombol **Logout**. Setelah diklik akan menjalankan atribut onTap yang berupa fungsi. Fungsi ini akan menghapus token pada storage, yaitu fungsi **logout()**

```
Future logout() async {  
  final SharedPreferences pref = await SharedPreferences.getInstance();  
  pref.clear();  
}
```

