LAPORAN PRAKTIKUM PEMROGRAMAN BERGERAK (MOBILE)



Dikerjakan Oleh:

Evi Fitriya (1201222005)

Fakultas Informatika (FIF)
Rekayasa Perangkat Lunak (S1)
SURABAYA
2024

DAFTAR ISI

| DAFT | TAR ISI | i |
|-------|---|----|
| GITHU | UB APLIKASI MOBILE | iv |
| TUGA | AS 1 | 1 |
| • | SOAL | 1 |
| • | Hasil Pengerjaan: | 1 |
| PERTI | EMUAN: 1 (Widget, Container, Row, Column) | 6 |
| • | Pembahasan: | 6 |
| • | Tampilan: | 7 |
| • | Code: | 7 |
| PERTI | EMUAN: 2 (GridView, PageView, Tab View) | 8 |
| • | Pembahasan: | 8 |
| • | Tampilan: | 10 |
| - | - Latihan1.dart: | 10 |
| - | pageView.dart: | 10 |
| - | - tabbar.dart: | 11 |
| • | Code: | 11 |
| Tugas | s 4 LMS | 12 |
| • | Soal | 12 |
| • | Pembahasan: | 12 |
| • | Tampilan: | 13 |
| • | Code: | 13 |
| PERTI | EMUAN: 3 (SnackBar, Dialog) | 14 |
| • | Pembahasan: | 14 |
| • | Tampilan: | 15 |
| • | Code: | 15 |
| PERTI | EMUAN: 4 | 16 |
| • | Pembahasan: | 16 |
| • | Tampilan: | 16 |
| • | Code: | 16 |
| PERTI | EMUAN 5 (Navigation) | 17 |
| • | Pembahasan: | 17 |

| • | Tampilan: | . 18 |
|-------|---------------------------------------|------|
| • | Code: | . 20 |
| TUGA | AS 5 LMS | . 21 |
| • | Pembahasan: | . 21 |
| • | Tampilan: | . 23 |
| • | Code: | . 23 |
| TUGA | AS 6 LMS | . 24 |
| • | Soal | . 24 |
| • | Pembahasan: | . 24 |
| • | Tampilan: | . 25 |
| • | Code: | . 25 |
| PERT | EMUAN 6 (Camera, Media) | . 26 |
| • | Pembahasan: | . 26 |
| • | Tampilan: | . 29 |
| • | Code: | . 32 |
| PERT | EMUAN 7 (Local Storage/Date) | . 33 |
| • | Pembahasan: | . 33 |
| • | Tampilan: | . 35 |
| • | Code: | . 35 |
| Perte | muan 8 dan 9 (ListView, CRUD SqfLite) | . 36 |
| • | Pembahasan: | . 36 |
| • | Tampilan: | . 37 |
| • | Code: | . 38 |
| Perte | muan 10 (Notification) | . 39 |
| • | Pembahasan: | . 39 |
| • | Tampilan: | . 41 |
| • | Code: | . 41 |
| Perte | muan 11 (Rest API) | . 42 |
| • | Pembahasan: | . 42 |
| • | Tampilan: | . 43 |
| • | Code: | . 44 |
| TEMP | PLATE | . 45 |
| • | Pembahasan: | . 45 |
| • | Tampilan: | . 45 |
| • | Code: | . 45 |

| LAMPIRA | N |
|---------|---------------------------|
| - | Main.dart |
| - | Meet1/latihan1.dart |
| - | Meet2/latihan1.dart50 |
| - | pageView.dart53 |
| - | tabbar.dart54 |
| - | button.dart57 |
| - | bot_navbar.dart62 |
| - | tugas4lms.dart64 |
| - | tugas5lms.dart67 |
| - | tugas6lms.dart69 |
| - | arguments_screen.dart |
| - | return_data_screen.dart77 |
| - | first_screen.dart79 |
| - | second_screen.dart81 |
| - | Media_camera.dart |
| - | User.dart84 |
| - | localData.dart85 |
| - | database_helper.dart |
| - | todo_db.dart |
| - | create_todo_widget.dart89 |
| - | todo_crud.dart |
| - | notifikasi.dart |
| - | todo_api_model.dart |
| - | api.dart |
| - | todo_api_crud.dart95 |
| - | view todo api.dart |

GITHUB APLIKASI MOBILE

https://github.com/evftrya/Mobile_Programming_Course/tree/main/praktikum_1

TUGAS 1

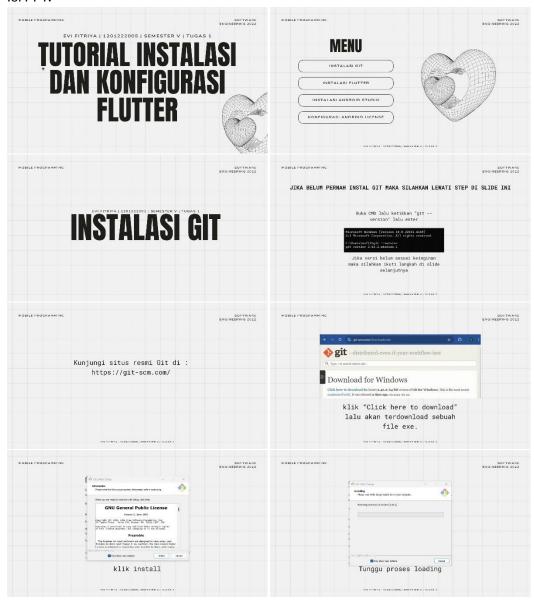
SOAL

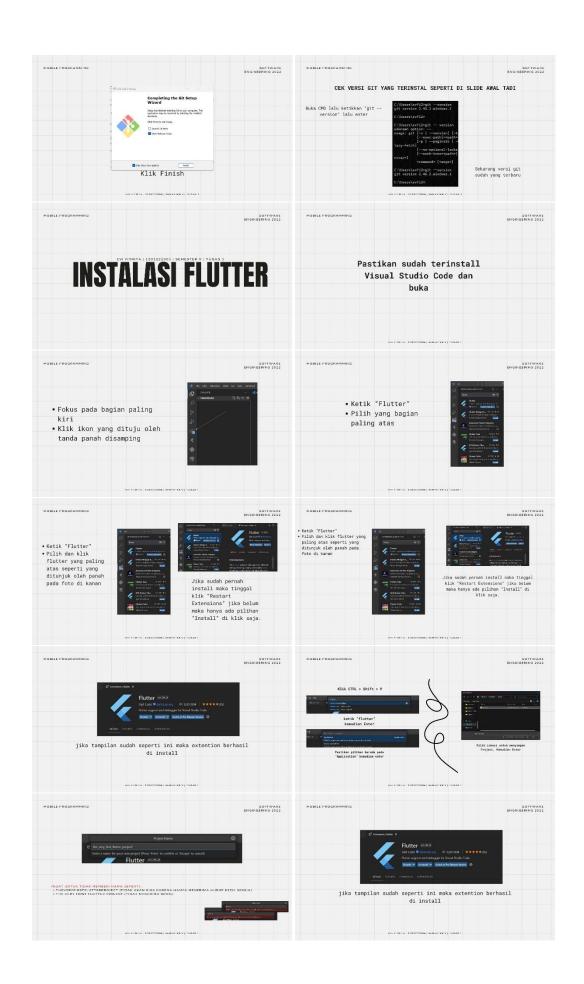
Membuat PPT terkait Instalasi dan Konfigurasi

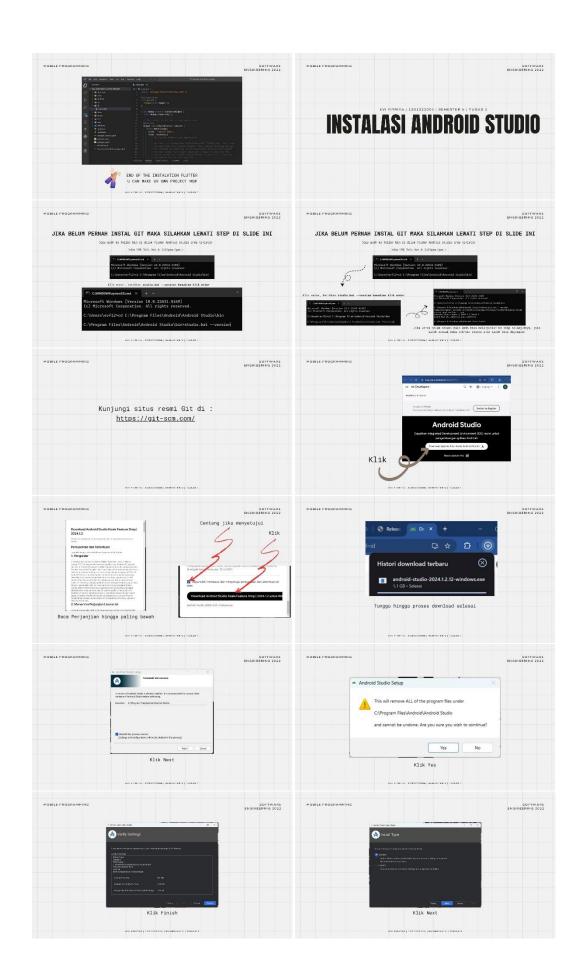
- Hasil Pengerjaan:
 - Link Canva:

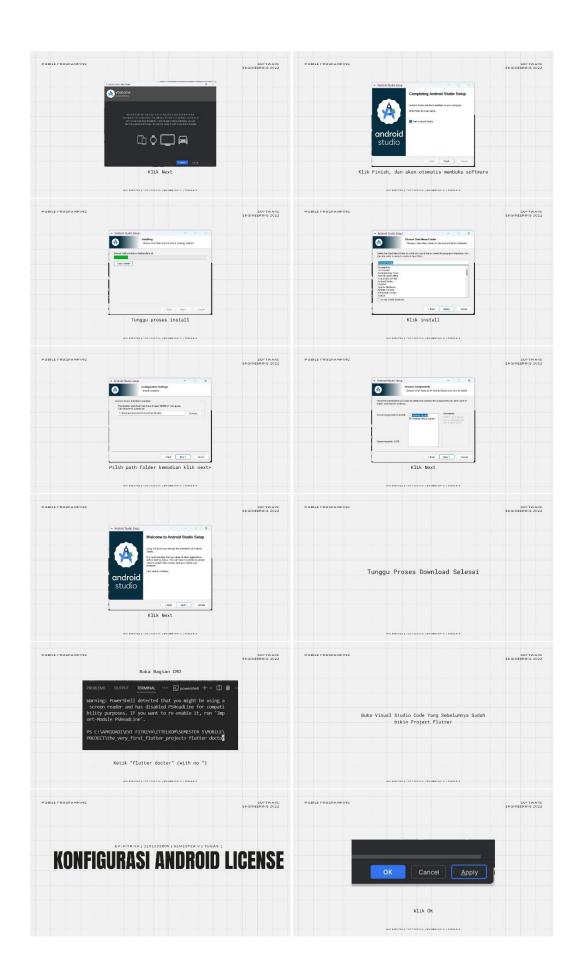
https://www.canva.com/design/DAGR4aJPnEE/sYIKUiqGD22_Y80m67wD9g/edit?ut m_content=DAGR4aJPnEE&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

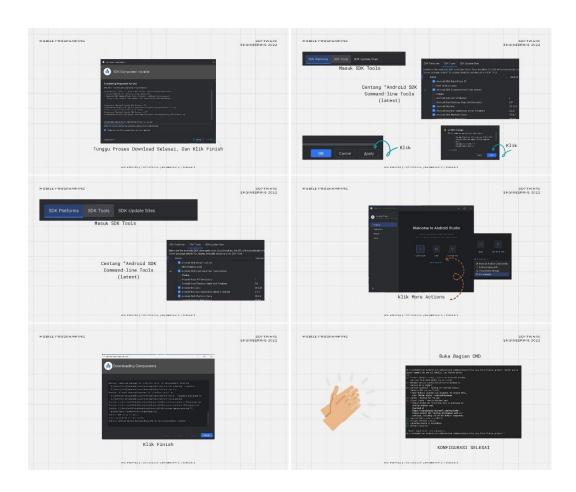
- Isi PPT:











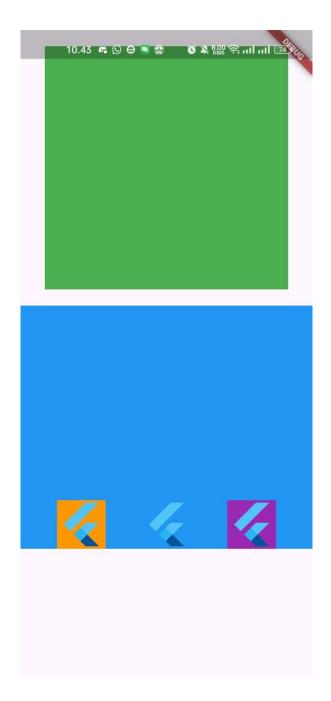
PERTEMUAN: 1 (Widget, Container, Row, Column)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "meet1/latihan1.dart" adalah file yang kiat tujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart

Isi file mencakup:

- Widget
 Setiap elemen yang ada didalam flutter seperti Text(), Button(), dll
- Kow
 Widget yang digunakan untuk mengatur elemen didalamnya agar berururtan menyamping.
- 3. Container: pembungkus
 - Untuk mengatur warna font, fontsize, dan segala macam gaya font menggunakan fungsi style()
 - Untuk mengatur color, padding, width, height, margin bisa lgsg saja didalam container tanpa embel-embel lagi.
- 4. Column: untuk mengatur tata letak agar vertikal terhadap yg lain
 - O Untuk mengatur "flex-direction" seperti di css menggunakan:
 - mainAxisAlignment: MainAxisAlignment.start (untuk rata kiri/atas (menyesuaikan tipe flexdirection))
 - mainAxisAlignment: MainAxisAlignment.end (untuk rata kanan/Bawah)
 - mainAxisAlignment: MainAxisAlignment.spaceBetween (untuk mengisi ruang dan menyesuaikan gap agar setara)
 - dan lain lain



♣ Code:

- Main.dart
- Latihan1.dart

PERTEMUAN: 2 (GridView, PageView, Tab View)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "pageView.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart
- File Dart "latihan1.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart
- File Dart "tabbar.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart

Isi File Mencakup:

- 1. Semua yang sudah dibahas di pembahasan Bab "Pertemuan 1" tentu diimplementasikan di Bab ini, jadi tidak perlu mengulang pembahasan, klik saja jika lupa.
- 2. Scaffold: sebagai pondasi dasar agar tertata elemen-elemennya
- 3. GridView.count : untuk membuat layout grid dan bisa diatur maksimal jumlah elemen didalamnya.
- 4. Untuk mengatur Margin/Padding menggunakan "EdgeInsects" Contoh:

```
margin: EdgeInsets.all(5),
artinya margin semua sisinya 5 px
margin: EdgeInsets.only(left:5, right:5),
```

Jika ingin hanya bagian tertentu bisa menggunakan "only"

- 5. Stack: digunakan untuk membuat susunan elemen menjadi tumpang tindih, mengikuti urutannya, misal urutan childrennya A-B-C, maka A ada dipaling bawah kemudian B kemudian C di paling atas.
- 6. PageView() digunakan untuk menavigasi halaman agar bisa digulir.
- 7. Icon: digunakan untuk mengambil data berupa ikon sesuai pemanggilannya.
- 8. DefaultTabController: digunakan untuk mengatur Tab di UI
 - o Memiliki atribut "length" yang harus di atur menyesuaikan banyak tabnya
- 9. AppBar: digunakan untuk membuar Tab yang ada di atas layar
- 10. TabBar: digunakan untuk membuat Area Tab
 - o Memiliki atribut "isScrollable" jika di true kan maka bisa di scroll
- 11. Tab(): digunakan untuk membuat Tab
- 12. TabBarView(): digunakan untuk mengatur isi dari Tab()
- 13. SafeArea(): digunakan untuk mengatur agar UI aktif selalu di bawah bilah notifikasi device apapun
- 14. Untuk memisahkan codingan menjadi funtion bisa seperti berikut: Function:

```
class Home extends StatelessWidget {
    const Home({
        super.key,
    });

    @override
    Widget build(BuildContext context) {
        return Text('hai');
    }
}
```

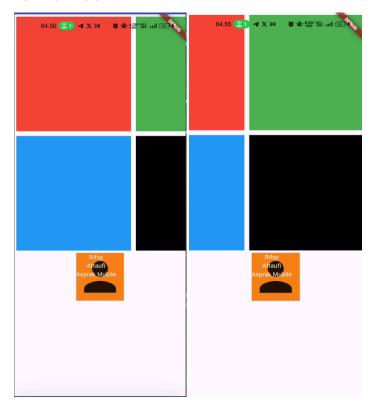
Cara panggilnya cukup panggil constraktor classnya saja, misal return Scaffold(
body: Container(
child: Home(),
),
);

Maka kode akan terbaca

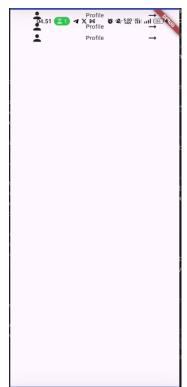
```
return Scaffold(
  body: Container(
   child: Text('hai'),
  ),
);
```

- 5. Row: untuk mengatur tata letak agar horizontal terhadap yg lain
 - o Untuk mengatur "flex-direction" seperti di css menggunakan:
 - mainAxisAlignment: MainAxisAlignment.start (untuk rata kiri/atas (menyesuaikan tipe flexdirection))
 - mainAxisAlignment: MainAxisAlignment.end (untuk rata kanan/Bawah)
 - mainAxisAlignment: MainAxisAlignment.spaceBetween (untuk mengisi ruang dan menyesuaikan gap agar setara)
 - dan lain lain

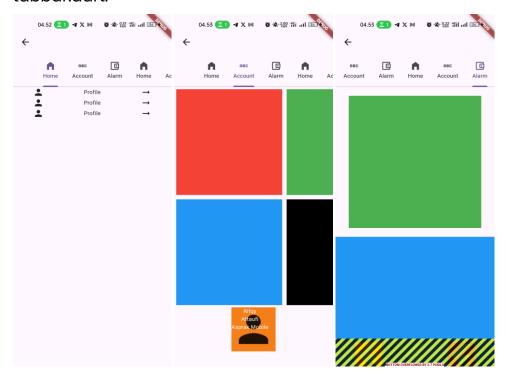
- Latihan1.dart:



- pageView.dart:



- tabbar.dart:

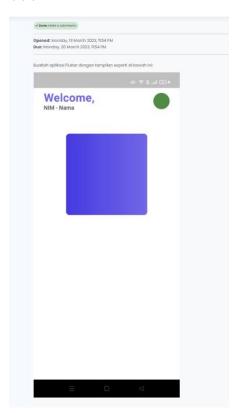


♣ Code:

- Main.dart
- Latihan1.dart
- pageView.dart
- tabbar.view

Tugas 4 LMS

Soal



Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "<u>Tugas4lms.dart</u>" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart berisi fokus utama.

Isi File Mencakup:

- Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1 dan Pertemuan 2
- 2. Agar widht 100% dari pembungkus dapat dilakukan seperti berikut:

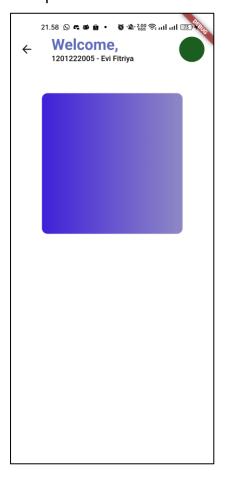
```
width: MediaQuery.of(context).size.width,
```

3. Agar warnanya bergradien:

4. Agar Shapenya berbentuk lingkaran (bisa juga lainnya)

shape: BoxShape.rectangle,

4 Tampilan:



♣ Code:

- Main.dart
- Tugas4lms.dart

PERTEMUAN: 3 (SnackBar, Dialog)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "button.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart berisi fokus utama pertemuan 3

Isi File Mencakup:

- 1. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1 dan Pertemuan 2
- 2. SizedBox(): digunakan untuk memberi ukruan tetap pada suatu elemen atau fungsi lainnya untuk memberikan spasi antar elemen
- 3. Dialog untuk memunculkan allert seperti pop up, menggunakan Allert Dialog().
- 4. TextButton(): Button yg berupa text
 - Untuk mengatur apabila diklik akan bagaimana, bisa mengisi di onPressed(){}
- 5. ElevatedButton(): Button yang berupa elemen dengan background berwarna yang bisa diberi text dan diwarnai sendiri
 - Untuk mengatur apabila diklik akan bagaimana, bisa mengisi di onPressed(){}
- 6. OutlinedButton(): Button yang berupa elemen dengan background berwarna putih dan memliki stroke dapat diisi text juga
 - Untuk mengatur apabila diklik akan bagaimana, bisa mengisi di onPressed(){}
- 7. IconButton(): Button yang berupa elemen icon
 - Untuk mengatur apabila diklik akan bagaimana, bisa mengisi di onPressed(){}
- 8. DropdownButton(): Button yang berupa dropdown
 - Untuk mengatur apabila diklik akan bagaimana, bisa mengisi di onPressed(){}
- 9. ScaffoldMessenger(): Pop up sementara di tengah layar untuk pemberitahuan
- 10. SnackBar(): Pop up sementara yg muncul di bawah layar untuk pemberitahuan



♣ Code:

- Main.dart
- Button.dart

PERTEMUAN: 4

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "Bot navbar.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart

Isi File Mencakup:

- 1. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1, Pertemuan 2, dan Pertemuan 3
- 2. PopupMenuButton(): untuk menampilkan popup dari suatu button jika diklik
- 3. PopupMenuItem(): tombol isi dari PopupMenuButton()
- 4. BottomNavigationBar(): area unguk navigasi yang ada dibawah
- 5. BottomNavigationBarItem(): TabBar yang ada dibawah , isi dari BottomNavigationBar().

4 Tampilan:



♣ Code:

- Main.dart
- Bot navbar.dart

PERTEMUAN 5 (Navigation)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "first_screen.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart dan akan mengarahkan pada halaman :
 - 1. Return data screen
 - 2. Second screen
 - 3. Arguments screen

Isi File Mencakup:

- 1. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1, Pertemuan 2, Pertemuan 3, dan Pertemuan 4
- 2. Navigator.push(): untuk menavigasikan halaman ke halaman lain dengan 2 parameter yakni context dan route. Route yg dimaksud disini adalah pemanggilan class lgsg dari file import lain.
- 3. Navigator.pushNamed(): untuk menavigasikan halaman ke halaman lain dengan 2 parameter yakni context dan route. Route yg dimaksud disini adalah pemanggilan string routenya
- 4. MaterialPageRoute(): untuk memberi transisi ketika menavigasikan ke halaman lain.
- Untuk membuat function untuk memecah kode menggunakan Future<void> nama function() async{ kode };

Untuk memanggil fungsi ini tinggal "_returnData()" saja

- 6. Navigator.pop(): untuk kembali ke halaman sebelumnya.
- 7. Untuk mengirim pesan dari file 1 ke file lain.
 - buat variable pesan

```
final String message;
```

- o jadikan konstraktor dari classnya membutuhkan variable pesan argumentsscreen extends statefulwidget {
 : ArgumentsScreen({super.key, required this.message});
- o pastikan ketika memanggil class, memberikan parameter pesan

o panggil dimanapun ketika diinginkan di class

```
children: [
    Text(widget.message),
    TextButton(
```

4 Tampilan:

- First_screen.dart



- Arguments_screen.dart



- Return_data_screen.dart



- Second_screen.dart



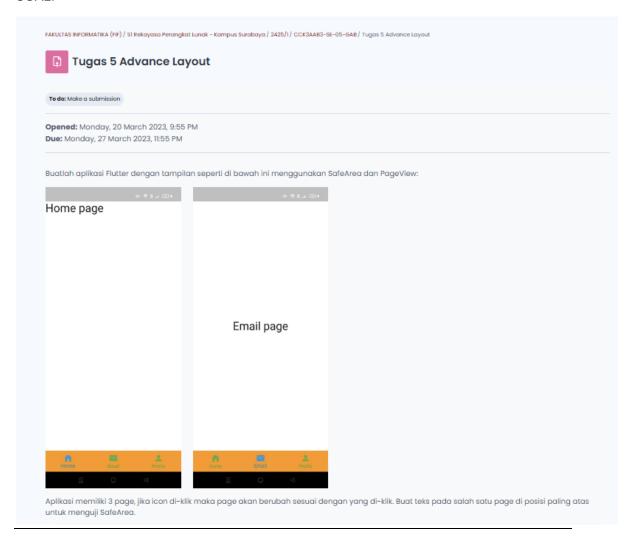
♣ Code:

- Main.dart
- arguments_screen.dart
- return_data_screen.dart
- <u>first_screen.dart</u>
- second_screen.dart

TUGAS 5 LMS

Pembahasan:

SOAL:



Komponen lainnya yang digunakan seperti yang sudah dijelaskan pada pertemuan – pertemuan sebelumnya, yang berbeda adalah sebagai berikut:

+ Penggunaan if else pada flutter dalam dart

Hal yang baru disini adalah ada 2 kondisi dimana dihalaman home tulisannya berada dikiri atas dan pada halaman lain tulisannya berada pada tengah halaman, itu menggunakan if else yang dimana "?" sebagai if dan ":" sebagai else

Contoh:

Apabila if terpenuhi maka kode akan seperti berikut:

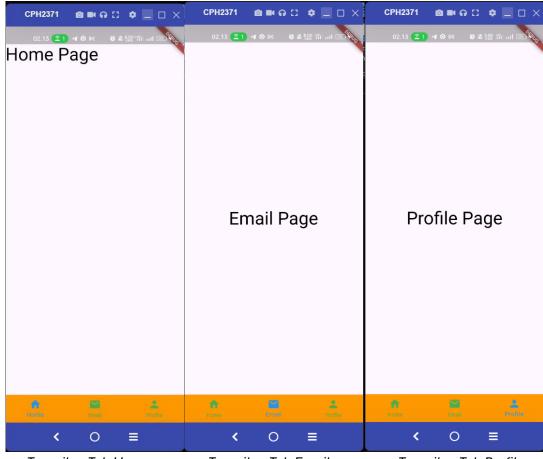
Sebaliknya apabila else terpenuhi kode akan seperti berikut:

→ Penggunaan saveArea

Save Area digunakan agar content didalam body berada dibawahnya top notification disemua device

Kodenya seperti berikut:





Tampilan Tab Home

Tampilan Tab Email

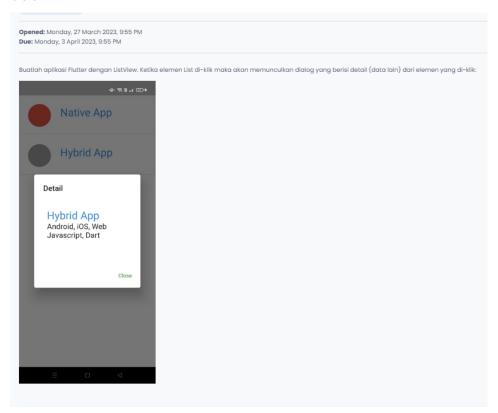
Tampilan Tab Profile

♣ Code:

- Main.dart
- Tugas5lms.dart

TUGAS 6 LMS

Soal

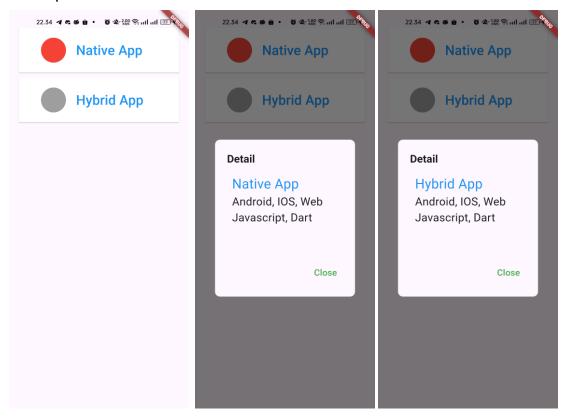


Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "tugas6lms.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart

Isi File Mencakup:

- Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1 Hingga Pertemuan 5
- 2. Untuk membuat pop up dialog menggunakan AlertDialog()
 - o Untuk mengatur bentuknya bisa menggunakan shape
 - Untuk membuatnya kotak bisa menggunakan RoundedRectangleBorder() kemudian atur borderRadiusnya dengan BorderRadius().
- 3. Tidak ada yang baru lagi ditugas ini selain yg disebutkan diatas.



♣ Code:

- Main.dart
- Tugas6lms.dart

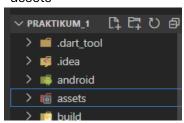
PERTEMUAN 6 (Camera, Media)

Pembahasan:

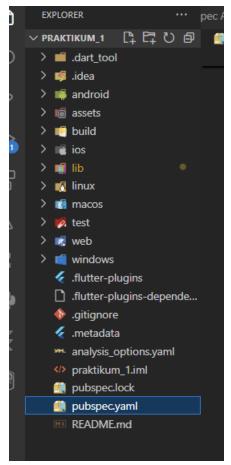
- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "media_camera.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart, disini berisi kode yang dapat menampilkan serta mengambil gambar baik dari galeri ataupun camera

Isi File Mencakup:

- 4. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1, Pertemuan 2, Pertemuan 3, Pertemuan 4, dan Pertemuan 5
- 5. Menggunakan 2 Dependencies yakni image_picker dan video_player
- 6. Cara pakai dependencies
 - ketik "flutter pub add image_picker video_player"
 - oflutter pub add image_picker video_player
 - o Kemudian enter
- 7. Untuk menampilkan Foto dari file yg sudah tersedia
 - Buat dulu Folder didalam tugasnya ya diluar folder "lib" dengan nama "assets"

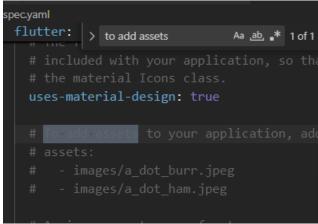


o Kemudian cari file "pubspec.yaml" ada dibarisan folder tugasnya lgsg



Klik file yg di tandai biru

Ctrl + F, masukkan text "To Add Assets"



Pada arahkan kursor text pada bagian "# assets:", klik ctrl+/

```
# To add assets to
assets:
# - images/a dot
```

Tambahkan "- assets/" (Sesuaikan sama nama folder) dibawahnya "assets:"

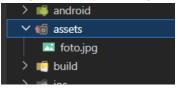
```
# To add assets to you
assets:
- assets/
# - images/a_dot_bur
# - images/a_dot_ham
```

o Isi folder assets dengan foto yang anda inginkan

- o Kembali ke file codingan anda
- o Pakai fungsi Image.asset(masukkan path fotonya disini)

```
--Image.asset("assets/foto.jpg"),
```

Dengan isi assets sebagai berikut:



- 8. Untuk menampilkan foto dari foto online
 - Copy link image online
 - o Kembali ke kondingan anda
 - o Pakai fungsi Image.network(paste link image foto online as string disini).

- 9. Untuk Mengakses Foto yang ada di gallery
 - o Pakai fungsi ImagePicker()
 - Pakai fungsi PickImage() yg ada didalam ImagePicker() menggunakan parameter "ImageSource.gallery"
 - o Pakai lagi variable "path"

10. Untuk Menampilkan foto yang sudah dipilih di gallery pakai function image.File(variable Image)

```
-if (ImageFile != null) Image.file(ImageFile!)
```

- Tampilan sebelum ditambah foto hasil online sebelum ditambah dari galery ataupun ambil langsung:



- Tampilan ketika pic image from galery:



- Tampilan ketika Take image:



- Hasil take video from galery:



- Hasil Take Video from Camera:



♣ Code:

- Main.dart
- Media_camera.dart

PERTEMUAN 7 (Local Storage/Date)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "localData.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart
- File "User.dart" adalah file untuk menyimpan data local storage agar bisa dikelola flutter

Isi File Mencakup:

- 1. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1, Pertemuan 2, Pertemuan 3, Pertemuan 4, Pertemuan 5, dan Pertemuan 6
- 2. Install lybrary "shared_prefferences"
 - o Buka terminal
 - Masukkan "flutter pub add shared_preferences"
 - o Enter

Shared Preferences digunakan untuk menyimpan data sederhana secara Lokal dalam bentuk key-value

Mencakup:

- SaveSharedPreference(): menyimpan nama pengguna (nama) ke SharedPreferences
- b. getSharedPreference(): membaca data nama pengguna yang telah disimpan di SharedPreferences.
- c. ClearData(): menghapus data nama pengguna yang ada di SharedPreferences
- 3. Install Lybrary flutter secure storage
 - o Buka Terminal
 - o Masukkan "flutter pub add future_secure

Flutter Secure Storage digunakan untuk menyimpan data yang lebih aman, seperti token autentikasi, dengan enkripsi, dumana memastikan data yang disimpan lebih aman karena data tersebut dienkripsi.

Mencakup:

- a. SaveLocalStorage(): menyimpan objek User (yang berisi username dan token) dalam bentuk JSON ke file lokal di perangkat
- b. GetLocalStorage(): membaca file JSON yang disimpan dan mengonversinya kembali menjadi objek User
- 4. Local File Storage
 - Masuk file pubspec.yaml yang ada di dalam project flutter di area setara folder "Lib"

Tambahkan "path_provider: ^2.0.11" pada dependencies:

```
# versions available, run `flutter pu
dependencies:
    flutter:
    sdk: flutter

pubspec.yaml

Version range of the package or SDK to use
Example:
    ^1.0.0

Source: pubspec.json

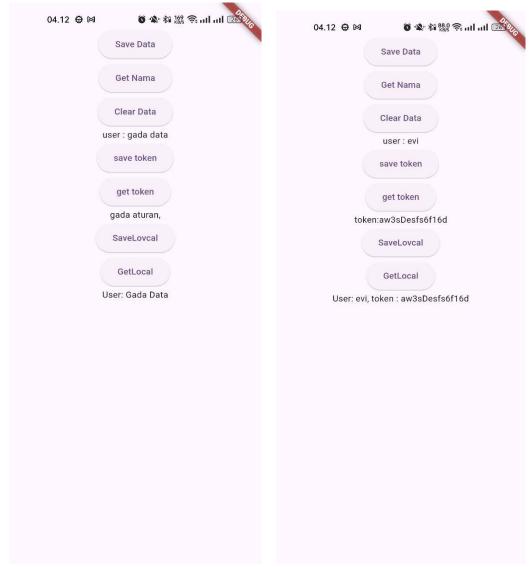
path_provider: ^2.1.5
sqflite: ^2.3.3+1
path: ^1.9.0
flutter_local_notifications: ^18.0.
http: ^1.2.2
```

- o Lalu ketik "flutter pub get" pada terminal
- o Enter

Local File Storage digunakan untuk menyimpan data dalam bentuk file. Di Flutter, kita bisa menggunakan path_provider untuk mendapatkan lokasi penyimpanan yang sesuai di perangkat dan menyimpan data dalam format file (misalnya JSON).

4 Tampilan:

- Ketika baru mulai kemudian setelah di get semua:



Code:

- Main.dart
- User.dart
- localData.dart

Pertemuan 8 dan 9 (ListView, CRUD SqfLite)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "todo_crud.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart
- File Dart "create_todo_widget" adalah file yang menyimpan pop up create dan edit
- File database_helper.dart adalah file untuk membantu menghubungkan database ke flutter
- File todo_db.dart berisi proses CRUD pada database

Isi File Mencakup:

- 5. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1, Pertemuan 2, Pertemuan 3, Pertemuan 4, Pertemuan 5, Pertemuan 6, dan Pertemuan 7
- 1. ListView digunakan untuk menampilkan view yang dapat diklik selayaknya button yang ketika diklik dapat memunculkan pop up.
- 2. Dependencies
 - a. Sqflite
 - b. Path

Cara mendapatkannya cukup run "flutter pub sqflite path" pada terminal

```
:ikum_1> flutter pub sqflite path
```

Pastikan pada pubspec.yaml ada "sqflite":

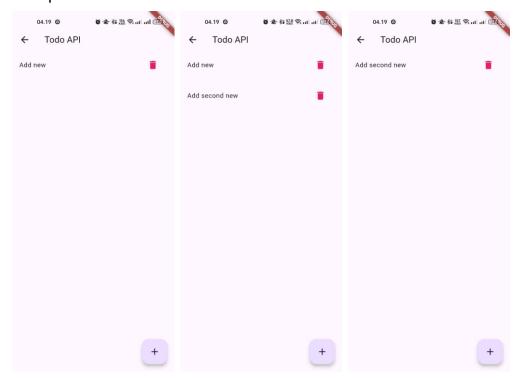
```
# versions available, run 'flutter pub out
dependencies:
    flutter:
        sdk: flutter

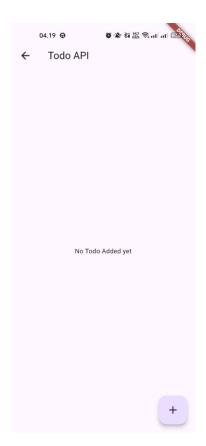
# The following adds the Cupertino Icons
# Use with the CupertinoIcons class for
cupertino_icons: ^1.0.6
image_picker: ^1.1.2
video_player: ^2.9.2
shared_preferences: ^2.3.3
flutter_secure_storage: ^9.2.2
path_provider: ^2.1.5
sqflite: ^2.3.3+1
path: ^1.9.0
flutter_local_notifications: ^18.0.1
http: ^1.2.2
```

3. Penjelasan Alur

- a. TodoCrud : menampilkan daftar todos dan memungkinkan penambahan, pengeditan, dan penghapusan todos.
- b. CreateTodoWidget: Dialog untuk menambah atau mengedit todo dengan form input.
- c. TodoDb (Database helper): untuk mengelola operasi CRUD di SQLite database, seperti menambah, mengupdate, menghapus, dan mengambil todos.
- d. DatabaseHelper: Mengelola inisialisasi database SQLite dan memastikan hanya ada satu instance database.
- e. TodoModel: model untuk menyimpan data sementara pada projek.

4 Tampilan:





♣ Code:

- Main.dart
- Database_helper.dart
- Todo_db.dart
- Create todo widget.dart
- Todo_crud.dart

Pertemuan 10 (Notification)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "<u>Bot navbar.dart</u>" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart

Isi File Mencakup:

- 1. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1, Pertemuan 2, Pertemuan 3, Pertemuan 4, Pertemuan 5, Pertemuan 6, Pertemuan 7, dan Pertemuan 8 juga 9
- 2. Pasang hal berikut:

- a. Copy teks diatas
- b. Cari folder Android>app>src>main>AndroidManifest.xml
- c. Paste pada lokasi berikut:

```
android:name="$(applicationName)"
android:name="$(applicationName)"
android:icon="@mipmap/ic_launcher">

<receiver android:exported="false" android:name="com.dexterous.flutterlocalnotifications.5cheduledNotificationReceiver" />
<receiver android:exported="false" android:name="com.dexterous.flutterlocalnotifications.5cheduledNotificationBootReceiver">

<intent-filter>

<action android:name="android.intent.action.BOOT_COMPLETED"/>
<action android:name="android.intent.action.MY_PACKAGE_REPLACED"/>
<action android:name="android.intent.action.QUICKBOOT_POWERON" />
</intent-filter>

</receiver>

<activity
android:name=".MainActivity"
android:name=".mainame"
and
```

Tujuannya adalah digunakan untuk mengonfigurasi aplikasi Flutter agar notifikasi terjadwal tetap berfungsi setelah perangkat di-restart atau setelah aplikasi di-update.

- 3. Install Flutter Local Notification
 - a. Kunjungi https://pub.dev/packages/flutter_local_notifications/install
 - b. salin kode berikut:

```
With Flutter:

$ flutter pub add flutter_local_notifications
```

- c. paste pada terminal anda
- d. enter
- e. kemudian ketik pada terminal => "flutter pub get"
- f. enter

g. pastikan di dependecies sudah terdaftar seperti berikut:

```
# the latest version available on publication.
# versions available, run `flutter pub outdated`.
dependencies:
    flutter:
        sdk: flutter

# The following adds the Cupertino Icons font to your at the second cupertino_icons: ^1.0.6
    image_picker: ^1.0.6
    image_picker: ^1.1.2
    video_player: ^2.9.2
    shared_preferences: ^2.3.3
    flutter_secure_storage: ^9.2.2
    path_provider: ^2.1.5
    sqflite: ^2.3.3+1
    path: ^1.9.0
    flutter_local_notifications: ^18.0.1
```

4. Alur File:

- a. FlutterLocalNotificationPlugin : Menyediakan API untuk menampilkan notifikasi.
- b. init(): Menginisialisasi plugin dan meminta izin untuk notifikasi di Android.
- c. showNotification(): Menampilkan notifikasi dengan judul, tubuh, dan payload yang diberikan.
- d. onDidReceiveNotification(): Digunakan untuk menangani interaksi pengguna dengan notifikasi, seperti navigasi ke screen tertentu berdasarkan payload.
- 5. Hal yang perlu ditambah di Main.dart
 - a. final GlobalKey<NavigatorState> navKey = GlobalKey<NavigatorState>();

```
import 'package:praktikum_1/meet89/todo_crud.dart';
// import 'package:praktikum_1/meet89/TodoCrud';

final GlobalKey<NavigatorState> navKey = GlobalKey<NavigatorState>();

Run|Debug|Profile
Future
```

Digunakan untuk mengontrol navigasi di aplikasi secara global. Ini memungkinkan Anda melakukan navigasi (seperti pindah halaman) dari mana saja dalam aplikasi, tanpa bergantung pada BuildContext.

b. idgetsFlutterBinding.ensureInitialized(); await Notifikasi.init();

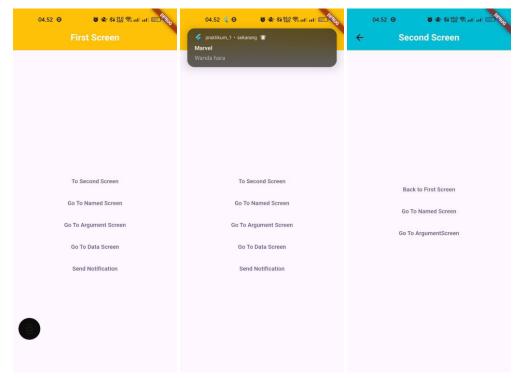
Memastikan semua pengaturan dan dependensi (seperti notifikasi) sudah diinisialisasi sebelum aplikasi berjalan. Ini penting untuk konfigurasi yang membutuhkan inisialisasi lebih dulu.

c. navigatorKey: navKey,

Menghubungkan navKey dengan MaterialApp, sehingga memungkinkan kontrol navigasi melalui navKey di seluruh aplikasi, bahkan di luar widget atau konteks.

6. Pastikan run pada smartphone asli a.k.a bukan emulator karena jika pada emulator, jarang muncul perizinan notifikasinya.

4 Tampilan:



♣ Code:

- Main.dart
- Notifikasi.dart

Pertemuan 11 (Rest API)

Pembahasan:

- File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
- File Dart "todo_api_crud" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart

Isi File Mencakup:

- 7. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada Pertemuan 1, Pertemuan 2, Pertemuan 3, Pertemuan 4, Pertemuan 5, Pertemuan 6, Pertemuan 7, Pertemuan 8 juga 9, dan Pertemuan 10
- 1. Pasang dependencies http
 - a. Buka terminal
 - b. Ketik "flutter pub add http"
 - c. Enter
 - d. Pastikan dependencies terdaftar pada file pubspec.yaml, seperti berikut

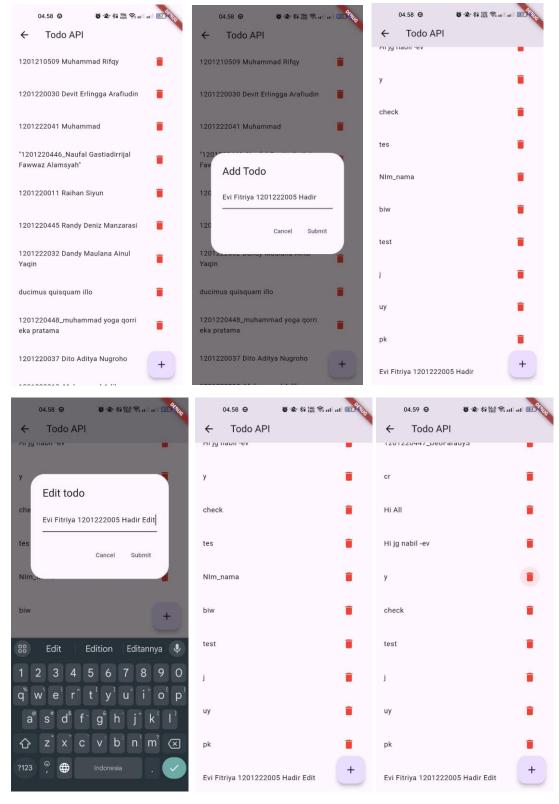
```
# versions available, run `flutter pub outdat
dependencies:
    flutter:
        sdk: flutter

# The following adds the Cupertino Icons for
# Use with the CupertinoIcons class for iOS
cupertino_icons: ^1.0.6
image_picker: ^1.1.2
video_player: ^2.9.2
shared_preferences: ^2.3.3
flutter_secure_storage: ^9.2.2
path_provider: ^2.1.5
sqflite: ^2.3.3+1
path: ^1.9.0
flutter_local_notifications: ^18.0.1
http: ^1.2.2
```

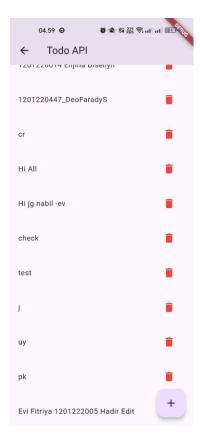
- e. Lakukan "flutter pub get" pada terminal
- f. enter

digunakan untuk melakukan HTTP request dari aplikasi Flutter ke server API. Dengan http, projek dapat mengirimkan data ke API (seperti POST atau PUT) dan mengambil data dari API (seperti GET).

Tampilan:



Y dihapus:



♣ Code:

- Main.dart
- Todo api model.dart
- Api.dart
- Todo_api_crud.dart
- View_todo_api.dart

TEMPLATE

- Pembahasan:
 - File Dart "main.dart" adalah file utama untuk menampilkan ke tampilan layar
 - File Dart "Bot navbar.dart" adalah file yang ditujukan untuk ditampilkan dilayar yang harus di import dulu di main.dart

Isi File Mencakup:

- 11. Elemen yang digunakan pada tugas ini beberapa sudah dijelaskan pada <u>Pertemuan 1, Pertemuan 2, dan Pertemuan 3</u>
- **4** Tampilan:
- Code:

Tinggal diklik saja untuk menuju ke kode (Semua Kode ada di bagian "Lampiran")

- Main.dart

LAMPIRAN

- Main.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_1/API/todo_api_crud.dart';
import 'package:praktikum_1/API/view_todo_Api.dart';
import 'package:praktikum 1/Notifikasi/notifikasi.dart';
import 'package:praktikum 1/meet1/latihan1.dart';
import 'package:praktikum_1/meet2/latihan1.dart';
import 'package:praktikum_1/meet2/tabbar.dart';
import 'package:praktikum_1/meet2/pageView.dart';
import 'package:praktikum_1/meet3n4/button.dart';
import 'package:praktikum_1/meet4/bot_navbar.dart';
import 'package:praktikum_1/tugas/tugas4lms.dart';
import 'package:praktikum 1/tugas/tugas5lms.dart';
import 'package:praktikum_1/meet5/arguments screen.dart';
import 'package:praktikum_1/tugas/tugas6lms.dart';
import 'package:praktikum_1/meet6/media_camera.dart';
import 'package:praktikum_1/meet7/localData.dart';
import 'package:praktikum_1/meet5/arguments_screen.dart';
import 'package:praktikum 1/meet5/first screen.dart';
import 'package:praktikum 1/meet5/second screen.dart';
import 'package:praktikum_1/meet5/return_data_screen.dart';
import 'package:praktikum_1/meet89/todo_crud.dart';
final GlobalKey<NavigatorState> navKey = GlobalKey<NavigatorState>();
Future<void> main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Notifikasi.init();
  runApp(const MyApp());
class MyApp extends StatelessWidget {
 const MyApp({super.key});
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Praktikum',
      theme: ThemeData(
```

```
colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
       useMaterial3: true,
      ),
      initialRoute: '/todoApi',
      navigatorKey: navKey,
      routes: {
        '/botnav': (context) => const Botnavbar(),
        '/return-data': (context) => const ReturnDataScreen(),
        '/button': (context) => const ButtonScreenState(),
        '/tugas6': (context) => const tugas_6_lms(),
        '/meet7': (context) => localData(),
        '/tugas4': (context) => const Evi_1201222005s(),
        '/tugas5': (context) => const AdvanceLayout(),
        '/M1/l1': (context) => const latihan_2(),
        '/M1/pageView': (context) => const Page_View(),
        '/M1/tabbar': (context) => const tabbar(),
        '/media-camera': (context) => const mediaCamera(),
        '/': (context) => const FirstScreen(),
        '/second': (context) => const SecondScreen(),
        '/todo': (context) => const TodoCrud(),
        '/todoApi': (context) => const todoApiCrud(),
        '/arguments': (context) =>
            const ArgumentsScreen(message: "Hi Everyone"),
class MyHomePage extends StatefulWidget {
 const MyHomePage({super.key, required this.title});
 final String title;
 State<MyHomePage> createState() => _MyHomePageState();
class _MyHomePageState extends State<MyHomePage> {
 int counter = 0;
```

```
void _incrementCounter() {
  setState(() {
    _counter++;
  });
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      backgroundColor: Theme.of(context).colorScheme.inversePrimary,
      title: Text(widget.title),
    body: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          const Text(
          Text(
            '$ counter',
            style: Theme.of(context).textTheme.headlineMedium,
          ),
        ],
```

- Meetl/latihanl.dart

```
import 'package:flutter/material.dart';
class Evi 1201222005 extends StatefulWidget {
  const Evi_1201222005({super.key});
 @override
 State<Evi 1201222005> createState() => Evi 1201222005State();
class _Evi_1201222005State extends State<Evi_1201222005> {
 @override
 Widget build(BuildContext context) {
    return Scaffold(
        body: Column(
      children: [
        Container(
          color: Colors.green,
          width: 300,
          height: 300,
          margin: EdgeInsets.all(20),
          padding: EdgeInsets.all(20),
          alignment: FractionalOffset(0.5, 0.5),
          child: Text(
            "Evi 1201222005",
            style: TextStyle(
                color: Colors.green, fontSize: 24,
                fontWeight: FontWeight.bold,
            ),
          ),
        ),
        Container(
            color: Colors.blue,
```

```
height: 300,
           child: Row(
             mainAxisAlignment: MainAxisAlignment.spaceEvenly,
             crossAxisAlignment: CrossAxisAlignment.end,
             children: [
               Container(
                 color: Colors.orange,
                 child: FlutterLogo(
                   size: 60.0,
                 ),
               ),
               Container(
                 color: Colors.blue,
                 child: FlutterLogo(
                   size: 60.0,
                 ),
               ),
               Container(
                 color: Colors.purple,
                 child: FlutterLogo(
                   size: 60.0,
                 ),
               ),
             ],
           ))
    ],
  ));
}
```

- Meet2/latihan1.dart

```
import 'package:flutter/material.dart';

class latihan_2 extends StatelessWidget {
  const latihan_2({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Column()
```

```
children: [
 Container(
    width: double.infinity,
    height: 500,
    child: GridView.count(
      crossAxisCount: 2,
      scrollDirection: Axis.horizontal,
      children: [
        Container(
          width: 100,
          height: 100,
          color: Colors.red,
          margin: EdgeInsets.all(5),
        ),
        Container(
          width: 100,
          height: 100,
          color: Colors.blue,
          margin: EdgeInsets.all(5),
        ),
        Container(
          width: 100,
          height: 100,
          color: Colors.green,
          margin: EdgeInsets.all(5),
        ),
        Container(
          width: 100,
          height: 100,
          color: Colors.black,
          margin: EdgeInsets.all(5),
      ],
    ),
  ),
 Container(
    child: Stack(
      children: [
        Container(
          width: 100,
          height: 100,
          color: Colors.yellow[900],
```

```
),
                 Icon(
                   Icons.person,
                   size: 100,
                 ),
                 Container(
                   child: Column(
                     children: [
                       Text(
                         "Rifqy",
                         style: TextStyle(color: Colors.white),
                       ),
                       Text(
                         "Attaufi",
                         style: TextStyle(color: Colors.white),
                       ),
                       Text(
                         "Asprak Mobile",
                         style: TextStyle(color: Colors.white),
                       ),
                     ],
                   ),
               ],
             ),
);
}
         ],
```

- pageView.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_1/meet2/latihan1.dart';
import 'package:praktikum_1/meet2/tabbar.dart';

class Page_View extends StatelessWidget {
   const Page_View({super.key});

   @override
   Widget build(BuildContext context) {
     return Scaffold(
       body: PageView(
       children: [Home(), latihan_2()],
      ),
     );
   }
}
```

- tabbar.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/widgets.dart';
import 'package:praktikum 1/meet2/latihan1.dart';
import 'package:praktikum_1/meet1/latihan1.dart';
class tabbar extends StatelessWidget {
  const tabbar({super.key});
  @override
  Widget build(BuildContext context) {
    return DefaultTabController(
      length: 6,
      child: Scaffold(
        appBar: AppBar(
          bottom: TabBar(isScrollable: true, tabs: [
              icon: Icon(Icons.home filled),
              text: "Home",
            ),
            Tab(
              icon: Icon(Icons.abc),
              text: "Account",
            ),
            Tab(
              icon: Icon(Icons.account balance wallet outlined),
              text: "Alarm",
            ),
            Tab(
              icon: Icon(Icons.home_filled),
              text: "Home",
            ),
            Tab(
              icon: Icon(Icons.abc),
              text: "Account",
            ),
            Tab(
              icon: Icon(Icons.account balance wallet outlined),
              text: "Alarm",
            ),
```

```
),
        body: SafeArea(
          child: TabBarView(
            children: [
              Home(),
              latihan 2(),
              Evi 1201222005(),
              Home(),
              latihan 2(),
              Evi 1201222005()
            ],
     ),
 });
class Home extends StatelessWidget {
  const Home({
    super.key,
  });
 @override
 Widget build(BuildContext context) {
    return Column(
      children: [
        Row(
          mainAxisAlignment: MainAxisAlignment.spaceAround,
          children: [
            Icon(Icons.person),
            Text("Profile"),
            Icon(Icons.arrow_right_alt),
          ],
        ),
        Row(
          mainAxisAlignment: MainAxisAlignment.spaceAround,
          children: [
            Icon(Icons.person),
            Text("Profile"),
            Icon(Icons.arrow_right_alt),
```

```
),
Row(
mainAxisAlignment: MainAxisAlignment.spaceAround,
children: [
    Icon(Icons.person),
    Text("Profile"),
    Icon(Icons.arrow_right_alt),
    ],
    )
    ],
    );
}
```

- button.dart

```
import 'package:flutter/material.dart';
class ButtonScreenState extends StatefulWidget {
  const ButtonScreenState({super.key});
 @override
  State<ButtonScreenState> createState() =>
 ButtonScreenStateState();
class ButtonScreenStateState extends State<ButtonScreenState> {
  String _selectedItem = 'Item 1';
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            SizedBox(
              height: 20,
            ),
            TextButton(
              style: TextButton.styleFrom(
                backgroundColor: Colors.blue,
              ),
              onPressed: () {
                ScaffoldMessenger.of(context).showSnackBar(
                  SnackBar(
                    duration: Duration(seconds: 2),
                    action: SnackBarAction(label: 'Detail',
onPressed: () {}),
                    content: Text(
                      'Pesan Berhasil',
                      style: TextStyle(
                        color: Colors.cyanAccent,
                      ),
```

```
),
                );
              },
              child: Text(
                 'Text Button x Snack Bar',
                style: TextStyle(
                  fontSize: 24,
                  color: Colors.white,
                ),
              ),
            ),
            ElevatedButton(
              style: TextButton.styleFrom(
                backgroundColor: Colors.black,
              ),
              onPressed: () {
                showDialog(
                   context: context,
                   builder: (context) {
                     return AlertDialog(
                       title: Text(
                         'Hapus Data',
                         style: TextStyle(fontWeight:
FontWeight.bold),
                       ),
                       content: Text('r u sure wanna delete this
data?'),
                       actions: [
                         TextButton(
                           onPressed: () {
                             Navigator.pop(context);
                           },
                           child: Text(
                             'No',
                             style: TextStyle(
                               color: Colors.white,
                             ),
                           ),
                           style:
                               TextButton.styleFrom(backgroundColor:
Colors.red),
```

```
),
                         TextButton(
                           onPressed: () {
                             Navigator.pop(context);
                           },
                           child: Text(
                             'Sure',
                             style: TextStyle(
                               color: Colors.white,
                             ),
                           ),
                           style: TextButton.styleFrom(
                               backgroundColor: Colors.green),
                         )
                       ],
                     );
                  },
                 );
              },
              child: Text(
                 "Elevated Button x Allert Dialog",
                style: TextStyle(
                   color: Colors.white,
                ),
              ),
            ),
            OutlinedButton(
              onPressed: () {
                 showDialog(
                   context: context,
                   builder: (context) {
                     return SimpleDialog(
                       title: Text(
                         'Pilih Metode',
                         style: TextStyle(fontWeight:
FontWeight.bold),
                       ),
                       children: [
                         SimpleDialogOption(
                           child: Text('BCA'),
                           onPressed: () {},
```

```
SimpleDialogOption(
              onPressed: () {},
              child: Text('BRI'),
            SimpleDialogOption(
              onPressed: () {},
              child: Text('BSI'),
            ),
          ],
        );
      },
    );
  },
  child: Text(
    "Outlined Button x Simple Dialog",
    style: TextStyle(
      color: const Color.fromARGB(255, 96, 30, 30),
    ),
  ),
),
IconButton(
  onPressed: () {},
  icon: Icon(Icons.thumb_up),
DropdownButton<String>(
    value: _selectedItem,
    items: [
      DropdownMenuItem(
        value: 'Item 1',
        child: Text('Item 1'),
      ),
      DropdownMenuItem(
        value: 'Item 2',
        child: Text('Item 2'),
      DropdownMenuItem(
        value: 'Item 3',
        child: Text('Item 3'),
      ),
    ],
    onChanged: (value) {
      setState(() {
```

```
_selectedItem = value!;
});
})

],
),
floatingActionButton: FloatingActionButton(
    onPressed: () {},
    child: Icon(Icons.add),
),
    //FloatingActBut bisa tanpa body karena setara body
);
}
```

bot_navbar.dart

```
import 'dart:ffi';
import 'package:flutter/material.dart';
import 'package:praktikum_1/meet3n4/button.dart';
import 'package:praktikum 1/meet2/tabbar.dart';
class Botnavbar extends StatefulWidget {
  const Botnavbar({super.key});
 @override
 State<Botnavbar> createState() => _BotnavbarState();
class _BotnavbarState extends State<Botnavbar> {
  int selectedindex = 0;
  static const List<Widget> halaman = <Widget>[
    ButtonScreenState(),
    tabbar(),
    Text(
      'Profile',
    ),
  1;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Latihan Flutter'),
        backgroundColor: Colors.blue,
        centerTitle: true,
        actions: [
          PopupMenuButton(
            icon: const Icon(
              Icons.more vert sharp,
              color: Colors.white,
            ),
            itemBuilder: (context) {
              return [
                PopupMenuItem(
```

```
onTap: () {},
                 child: const Text('Logout'),
               ),
             ];
           },
         )
       ],
     ),
     body: Center(
       child: halaman.elementAt(_selectedindex),
     ),
     bottomNavigationBar: BottomNavigationBar(
       items: const [
         BottomNavigationBarItem(
           label: 'Home',
           icon: Icon(Icons.home),
         ),
         BottomNavigationBarItem(
           label: 'History',
           icon: Icon(Icons.history),
         ),
         BottomNavigationBarItem(
           label: 'Profile',
           icon: Icon(Icons.person),
         ),
       ],
       currentIndex: _selectedindex,
       onTap: (t) {
         setState(() {
           _selectedindex = t;
         });
    ),
});
```

- tugas4lms.dart

```
import 'package:flutter/material.dart';
class Evi 1201222005s extends StatelessWidget {
  const Evi_1201222005s({super.key});
  @override
  Widget build(BuildContext context) {
    return new Scaffold(
      appBar: new AppBar(
        backgroundColor: Colors.white,
        title: DefaultTextStyle(
          style: TextStyle(),
          child: new Row(
            children: [
              new Container(
                color: Colors.white,
                child: new Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    Text(
                       "Welcome,",
                      style: TextStyle(
                           color: const Color.fromARGB(255, 96, 108,
186),
                          fontWeight: FontWeight.bold,
                          fontSize: 30),
                    ),
                    Text(
                       "1201222005 - Evi Fitriya",
                      style: TextStyle(
                           color: Colors.black,
                          fontSize: 15,
                          fontWeight: FontWeight.w500),
                    ),
                  ],
                ),
              ),
              Spacer(),
              new Container(
                width: 45.0,
```

```
height: 45.0,
          decoration: BoxDecoration(
            color: Colors.green[900],
            shape: BoxShape.circle,
          ),
        ),
      ],
    ),
  ),
),
body: new Container(
  color: Color.fromARGB(255, 255, 255, 255),
  width: MediaQuery.of(context).size.width,
  child: new Column(
    children: [
      new Container(
        margin: EdgeInsets.symmetric(vertical: 50.0),
        color: Colors.white,
        child: new Row(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            new Container(
              width: 250.0,
              height: 250.0,
              decoration: BoxDecoration(
                  gradient: LinearGradient(
                    colors: [
                      Color.fromARGB(255, 63, 32, 217),
                      Color(0xFF8F89C5)
                    ],
                    begin: Alignment.centerLeft,
                    end: Alignment.centerRight,
                  ),
                  shape: BoxShape.rectangle,
                  borderRadius: BorderRadius.circular(10.0)),
            )
          ],
        ),
     )
```

```
);
}
}
```

- tugas5lms.dart

```
import 'package:flutter/material.dart';
class AdvanceLayout extends StatefulWidget {
  const AdvanceLayout({super.key});
 @override
  State<AdvanceLayout> createState() => _AdvanceLayout();
class _AdvanceLayout extends State<AdvanceLayout> {
  int _selectIndex = 0;
  static const List<Widget> pags = <Widget>[
    Text(
      'Home Page',
    ),
    Text(
      'Email Page',
    ),
    Text(
      'Profile Page',
    ),
  1;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: DefaultTextStyle(
            style: TextStyle(
              color: Colors.black,
              fontSize: 40,
            ),
            child: _selectIndex == 0
                ? Column(children: [
                    pags.elementAt( selectIndex),
                  ])
                : Center(
                    child: pags.elementAt(_selectIndex),
```

```
),
  bottomNavigationBar: BottomNavigationBar(
    items: const [
      BottomNavigationBarItem(
        label: 'Home',
        icon: Icon(Icons.home),
      ),
      BottomNavigationBarItem(
        label: 'Email',
        icon: Icon(Icons.email),
      ),
      BottomNavigationBarItem(
        label: 'Profile',
        icon: Icon(Icons.person),
      ),
    ],
    backgroundColor: Colors.orange,
    unselectedItemColor: Colors.green,
    selectedItemColor: Colors.blue,
    currentIndex: _selectIndex,
    onTap: (indexSelected) {
      setState(() {
        _selectIndex = indexSelected;
      });
    },
  ),
);
```

- tugas6lms.dart

```
import 'package:flutter/material.dart';
class tugas 6 lms extends StatefulWidget {
  const tugas_6_lms({super.key});
 @override
  State<tugas_6_lms> createState() => _tugas_6_lmsState();
class _tugas_6_lmsState extends State<tugas_6_lms> {
 @override
 Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.start,
            children: [
              Container(
                margin: EdgeInsets.only(left: 20.0, right: 20.0),
                padding: EdgeInsets.all(0),
                child: ElevatedButton(
                  style: ElevatedButton.styleFrom(
                    shadowColor: Colors.grey,
                    backgroundColor: const Color.fromARGB(255, 255,
255, 255),
                    shape: RoundedRectangleBorder(
                      borderRadius: BorderRadius.zero,
                    ),
                  ),
                  child: Row(
                    children: [
                      Container(
                        width: 50.0,
                        height: 50.0,
                        margin: EdgeInsets.all(20.0),
                        decoration: BoxDecoration(
                          color: Colors.red,
                          shape: BoxShape.circle,
```

```
),
                      Text(
                         "Native App",
                         style: TextStyle(
                           color: Colors.blue,
                           fontSize: 30,
                         ),
                    ],
                   ),
                  onPressed: () {
                    showDialog(
                      context: context,
                      builder: (context) {
                         return AlertDialog(
                           shape: RoundedRectangleBorder(
                             borderRadius:
BorderRadius.all(Radius.circular(10)),
                           ),
                           title: Text(
                             'Detail',
                             style: TextStyle(fontWeight:
FontWeight.bold),
                           ),
                           content: Container(
                             width: MediaQuery.of(context).size.width
 40,
                             height: 150,
                             child: Container(
                               padding: EdgeInsets.only(left: 10),
                               child: Column(
                                 mainAxisAlignment:
MainAxisAlignment.start,
                                 children: [
                                   Container(
                                     width:
MediaQuery.of(context).size.width,
                                     child: Text(
                                        'Native App',
```

```
style: TextStyle(
                                          fontSize: 30,
                                          color: Colors.blue,
                                        ),
                                        textAlign: TextAlign.start,
                                      ),
                                    ),
                                    Container(
                                      width:
MediaQuery.of(context).size.width,
                                      child: Text(
                                        'Android, IOS, Web Javascript,
Dart',
                                        style: TextStyle(fontSize:
25),
                                      ),
                                    ),
                                 ],
                               ),
                             ),
                           ),
                           actions: [
                             TextButton(
                               onPressed: () {
                                 Navigator.pop(context);
                               },
                               child: Text(
                                  'Close',
                                 style: TextStyle(
                                      color: Colors.green, fontSize:
20),
                               ),
                             ),
                           ],
                         );
                       },
                    );
                  },
```

```
SizedBox(
                width: 20,
                height: 10,
              ),
              Container(
                margin: EdgeInsets.only(left: 20.0, right: 20.0),
                padding: EdgeInsets.all(0),
                child: ElevatedButton(
                   style: ElevatedButton.styleFrom(
                     shadowColor: Colors.grey,
                    backgroundColor: const Color.fromARGB(255, 255,
255, 255),
                    shape: RoundedRectangleBorder(
                      borderRadius: BorderRadius.zero,
                    ),
                   ),
                  child: Row(
                     children: [
                      Container(
                         width: 50.0,
                         height: 50.0,
                         margin: EdgeInsets.all(20.0),
                         decoration: BoxDecoration(
                           color: Colors.grey,
                           shape: BoxShape.circle,
                         ),
                       ),
                      Text(
                         "Hybrid App",
                         style: TextStyle(
                           color: Colors.blue,
                           fontSize: 30,
                         ),
                       )
                    ],
                   ),
                  onPressed: () {
                    showDialog(
                       context: context,
                      builder: (context) {
                         return AlertDialog(
                           shape: RoundedRectangleBorder(
```

```
borderRadius:
BorderRadius.all(Radius.circular(10)),
                           ),
                           title: Text(
                             'Detail',
                             style: TextStyle(fontWeight:
FontWeight.bold),
                           content: Container(
                             width: MediaQuery.of(context).size.width
 40,
                             height: 150,
                             child: Container(
                               padding: EdgeInsets.only(left: 10),
                               child: Column(
                                 mainAxisAlignment:
MainAxisAlignment.start,
                                 children: [
                                   Container(
                                     width:
MediaQuery.of(context).size.width,
                                     child: Text(
                                       'Hybrid App',
                                       style: TextStyle(
                                         fontSize: 30,
                                         color: Colors.blue,
                                         // backgroundColor:
Colors.red,
                                       ),
                                       textAlign: TextAlign.start,
                                     ),
                                   ),
                                   Container(
                                     width:
MediaQuery.of(context).size.width,
                                     child: Text(
                                       'Android, IOS, Web Javascript,
Dart'
```

```
style: TextStyle(fontSize:
 25),
                                    ),
                                  ),
                                ],
                              ),
                            ),
                           ),
                           actions: [
                            TextButton(
                              onPressed: () {
                                Navigator.pop(context);
                              },
                              child: Text(
                                'Close',
                                style: TextStyle(
                                    color: Colors.green, fontSize:
 20),
                              ),
                            ),
                        );
             },
),
),
),
),
),
}
```

- arguments_screen.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_1/meet5/first_screen.dart';
class ArgumentsScreen extends StatefulWidget {
  const ArgumentsScreen({super.key, required this.message});
 final String message;
 @override
  State<ArgumentsScreen> createState() =>
_ArgumentsScreenState();
class _ArgumentsScreenState extends State<ArgumentsScreen> {
 @override
 Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        centerTitle: true,
        title: const Text(
          "Arguments Screen",
          style: TextStyle(color: Colors.white, fontWeight:
FontWeight.bold),
        ),
        backgroundColor: Colors.blueGrey,
      ),
      body: Center(
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.center,
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            Text(widget.message),
            TextButton(
              onPressed: () {
                Navigator.pop(context,
                    MaterialPageRoute(builder: (context) =>
FirstScreen()));
              child: const Text('Back to First Screen'),
```

- return_data_screen.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_1/meet5/arguments_screen.dart';
import 'package:praktikum_1/meet5/second_screen.dart';
class ReturnDataScreen extends StatefulWidget {
  const ReturnDataScreen({super.key});
  @override
  State<ReturnDataScreen> createState() =>
_ReturnDataScreenState();
class _ReturnDataScreenState extends State<ReturnDataScreen>
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        centerTitle: true,
        title: const Text(
          "Return Data Screen",
          style: TextStyle(color: Colors.white, fontWeight:
FontWeight.bold),
        ),
        backgroundColor: Colors.black45,
      ),
      body: Center(
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.center,
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            TextButton(
              onPressed: () {
                Navigator.pop(context, 'Halo Dari Data
Screen');
              child: const Text('Go Back'),
            ),
          ],
```

```
- ),
- );
- }
- }
```

- first_screen.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_1/Notifikasi/notifikasi.dart';
import 'package:praktikum_1/meet5/arguments_screen.dart';
import 'package:praktikum_1/meet5/second_screen.dart';
class FirstScreen extends StatefulWidget {
  const FirstScreen({super.key});
  State<FirstScreen> createState() => _FirstScreenState();
class _FirstScreenState extends State<FirstScreen> {
 Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        centerTitle: true,
        title: const Text(
          "First Screen",
          style: TextStyle(color: Colors.white, fontWeight: FontWeight.bold),
        backgroundColor: Colors.amber,
      ),
      body: SafeArea(
        child: Container(
          padding: EdgeInsets.only(top: 20),
          margin: EdgeInsets.only(top: 40),
          child: Center(
            child: Column(
              crossAxisAlignment: CrossAxisAlignment.center,
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                TextButton(
                  onPressed: () {
                    Navigator.push(
                      context,
                      MaterialPageRoute(
                        builder: (context) => SecondScreen(),
                      ),
                  child: const Text('To Second Screen'),
                TextButton(
                  onPressed: () {
                    Navigator.pushNamed(context, '/second');
                  child: Text('Go To Named Screen'),
                TextButton(
                  onPressed: () {
                    Navigator.push(
                      context,
                      MaterialPageRoute(
```

```
builder: (context) => ArgumentsScreen(
                          message: 'Ini pesan dari first "Marvelous"'),
                child: Text('Go To Argument Screen'),
              TextButton(
                onPressed: () {
                 _returnData();
                child: Text('Go To Data Screen'),
              TextButton(
               onPressed: () {
                 Notifikasi.showNotification(
                      title: "Marvel", body: "Wanda hara", payload: "second");
                child: Text('Send Notification'),
Future<void> _returnData() async {
 final result = await Navigator.pushNamed(context, '/return-data');
 ScaffoldMessenger.of(context).showSnackBar(
   SnackBar(
     content: Text(
        result.toString(),
```

- second_screen.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_1/meet5/first_screen.dart';
class SecondScreen extends StatefulWidget {
  const SecondScreen({super.key});
 @override
 State<SecondScreen> createState() => _SecondScreenState();
class SecondScreenState extends State<SecondScreen> {
 @override
 Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        centerTitle: true,
        title: const Text(
          "Second Screen",
          style: TextStyle(color: Colors.white, fontWeight:
FontWeight.bold),
        ),
        backgroundColor: Colors.cyan,
      ),
      body: Center(
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.center,
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            TextButton(
              onPressed: () {
                Navigator.pop(context,
                    MaterialPageRoute(builder: (context) =>
FirstScreen()));
              child: const Text('Back to First Screen'),
            ),
            TextButton(
                onPressed: () {
                  Navigator.pushNamed(context, '/');
```

- Media_camera.dart

```
import 'dart:ffi';
import 'dart:io';
import 'package:flutter/material.dart';
import 'package:image_picker/image_picker.dart';
import 'package:video_player/video_player.dart';
class mediaCamera extends StatefulWidget {
 const mediaCamera({super.key});
  @override
  State<mediaCamera> createState() => mediaCameraState();
class _mediaCameraState extends State<mediaCamera> {
  Widget build(BuildContext context) {
   return Scaffold(
      appBar: AppBar(
        title: const Text(
          'Media Camera',
      body: SingleChildScrollView(
        child: Center(
          child: Container(
            child: Column(
              children: [
                Image.asset("assets/foto.jpg"),
                Image.network(
                    "https://i.pinimg.com/736x/e3/07/85/e30785754f110c9ab5713d06abb7fed3.jp
g"),
                ElevatedButton(
                 onPressed: () {
```

```
pickImage();
                },
                child: Text('Pick Image Form Gallery'),
              ElevatedButton(
                onPressed: () {
                  takePicture();
                child: Text('Take Image Form Gallery'),
              ElevatedButton(
                onPressed: () {
                  pickVideo();
                child: Text('Pick Video Form Gallery'),
              ElevatedButton(
                onPressed: () {
                  takeVideo();
                child: Text('Take Video Form Gallery'),
              if (ImageFile != null) Image.file(ImageFile!),
              if (VideoFile != null)
                AspectRatio(
                  aspectRatio: controller!.value.aspectRatio,
                  child: VideoPlayer(_controller!),
   ),
File? ImageFile;
Future pickImage() async {
  final ImageFile =
      await ImagePicker().pickImage(source: ImageSource.gallery);
  if (ImageFile == null) return;
 final imageTemp = File(ImageFile.path);
  setState(() {
    this.ImageFile = imageTemp;
 });
Future takePicture() async {
  final ImageFile = await ImagePicker().pickImage(source: ImageSource.camera);
  if (ImageFile == null) return;
  final ImageTemp = File(ImageFile.path);
  setState(() {
    this.ImageFile = ImageTemp;
```

```
File? VideoFile;
VideoPlayerController? _controller;
Future pickVideo() async {
  final VideoFile =
      await ImagePicker().pickVideo(source: ImageSource.gallery);
  if (VideoFile == null) return;
 final VideoTemp = File(VideoFile.path);
 _controller = VideoPlayerController.file(VideoTemp)
    ..initialize().then((_) {
      setState(() {
        this.VideoFile = VideoTemp;
     _controller!.play();
      _controller!.setLooping(true);
    });
Future takeVideo() async {
  final VideoFile = await ImagePicker().pickVideo(source: ImageSource.camera);
  if (VideoFile == null) return;
  final VideoTemp = File(VideoFile.path);
  _controller = VideoPlayerController.file(VideoTemp)
    ..initialize().then((_) {
      setState(() {
        this.VideoFile = VideoTemp;
      });
      _controller!.play();
      _controller!.setLooping(true);
    });
```

- User.dart

```
import 'package:flutter/foundation.dart';
import 'package:flutter/gestures.dart';

class User {
    String username;
    String token;
    User({required this.username, required this.token});

    Map<String, dynamic> tolson() => {
        'username': username,
        'token': token,
        };
    factory User.FromJson(Map<String, dynamic> json) {
        return User(
        username: json['username'],
        token: json['token'],
```

```
);
}
}
```

- localData.dart

```
import 'dart:convert';
import 'package:flutter/material.dart';
import 'package:praktikum_1/meet7/model/User.dart';
import 'package:flutter_secure_storage/flutter_secure_storage.dart';
import 'package:path provider/path provider.dart';
import 'package:shared_preferences/shared_preferences.dart';
class localData extends StatefulWidget {
 localData({super.key});
  State<localData> createState() => localDataState();
String nama = "evi";
String SavedName = "";
String token = "aw3sDesfs6f16d";
String SavedToken = "";
class localDataState extends State<localData> {
 @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Center(
          child: Column(
            children: [
              ElevatedButton(
                onPressed: () {
                  SaveSharedPreference();
                child: Text('Save Data'),
              ElevatedButton(
                onPressed: () {
                  getSharedPreference();
                child: Text('Get Nama'),
              ElevatedButton(
                onPressed: () {
                  ClearData();
                child: Text('Clear Data'),
              Text(SavedName.isNotEmpty
                  : "user : gada data"),
              ElevatedButton(
```

```
onPressed: () {
                SaveSecureStorage();
              child: Text("save token"),
            ElevatedButton(
              onPressed: () {
                getStorage();
              child: Text("get token"),
            Text(
                SavedToken.isNotEmpty ? "token:$SavedToken" : "gada aturan,"),
            ElevatedButton(
              onPressed: () {
                SaveLocalStorage();
              child: Text("SaveLovcal"),
            ElevatedButton(
              onPressed: () {
                GetLocalStorage();
              child: Text("GetLocal"),
            ),
            Text(LocalUser != null
                ? "User: ${LocalUser!.username}, token : ${LocalUser!.token}"
                : "User: Gada Data"),
   ),
void SaveSharedPreference() async {
  SharedPreferences prefs = await SharedPreferences.getInstance();
  prefs.setString("username", nama);
void getSharedPreference() async {
  SharedPreferences prefs = await SharedPreferences.getInstance();
  setState(() {
    SavedName = prefs.getString("username")!;
 });
void ClearData() async {
  SharedPreferences prefs = await SharedPreferences.getInstance();
  setState(() {
    SavedName = "";
  });
void SaveSecureStorage() async {
  final storage = FlutterSecureStorage();
  await storage.write(key: "usertoken", value: token);
```

```
void getStorage() async {
 final storage = FlutterSecureStorage();
 String? value = await storage.read(key: "usertoken");
 setState(() {
   SavedToken = value ?? "data data";
 });
void SaveLocalStorage() async {
 final filename = 'local_user';
 final file = File(
      '${(await getApplicationDocumentsDirectory()).path}/$filename.json');
 User user = User(username: nama, token: token);
 await file.writeAsString(jsonEncode(user.tolson()));
User? LocalUser;
Future<void> GetLocalStorage() async {
 final fileename = 'local user';
 final file = File(
      '${(await getApplicationDocumentsDirectory()).path}/$fileename.json');
 String fileContent = await file.readAsString();
  setState(() {
    LocalUser = User.FromJson(jsonDecode(fileContent));
  });
```

database_helper.dart

```
import 'package:path/path.dart';
import 'package:praktikum_1/meet89/db/todo_db.dart';
import 'package:sqflite/sqflite.dart';
// import 'package:praktikum_1/meet8/db/todo_db.dart';

class DatabaseHelper {
    Database? _database;

    Future<Database> get database async {
        if (_database != null) {
            return _database!;
        }

        _database = await _initDatabase();
        return _database!;
    }

    Future<String> get fullPath async {
        const name = 'todo.db';

        final path = await getDatabasesPath();
        return join(path, name);
    }

    Future<Database> _initDatabase() async {
```

```
final path = await fullPath;

var database = await openDatabase(path,
    version: 1, onCreate: create, singleInstance: true);

return database;
}

Future<void> create(Database db, int version) async {
    TodoDb todoDB = TodoDb();
    await todoDB.createTable(db);
}
```

- todo_db.dart

```
import 'package:praktikum_1/meet89/db/database_helper.dart';
import 'package:praktikum_1/meet89/model/todo_model.dart';
import 'package:sqflite/sqflite.dart';
class TodoDb {
 final tableName = 'todo';
 Future<void> createTable(Database db) async {
    await db.execute("""
       CREATE TABLE IF NOT EXISTS $tableName(
         "id" INTEGER NOT NULL,
          PRIMARY KEY("id" AUTOINCREMENT)
     """);
 Future<int> create({required String title}) async {
   final database = await DatabaseHelper().database;
   return await database.rawInsert("""
       INSERT INTO $tableName (title) VALUES (?)
      """, [title]);
 Future<List<TodoModel>> getAll() async {
   final database = await DatabaseHelper().database;
    final todos = await database.rawQuery("""
       SELECT * FROM $tableName
     """);
   return todos.map((todo) => TodoModel.fromSqfliteDatabase(todo)).toList();
 Future<TodoModel> getById(int id) async {
   final database = await DatabaseHelper().database;
    final todo = await database.rawQuery("""
       SELECT * FROM $tableName WHERE id=?
```

```
""", [id]);

return TodoModel.fromSqfliteDatabase(todo.first);
}

Future<int> update({required int id, String? title}) async {
  final database = await DatabaseHelper().database;

return await database.update(tableName, {if (title != null) 'title': title},
  where: 'id = ?',
  whereArgs: [id],
  conflictAlgorithm: ConflictAlgorithm.rollback);
}

Future<void> delete(int id) async {
  final database = await DatabaseHelper().database;
  await database.delete(tableName, where: 'id = ?', whereArgs: [id]);
}
```

- create_todo_widget.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/foundation.dart';
import 'package:praktikum 1/API/model/todo api model.dart';
import 'package:praktikum 1/meet89/model/todo model.dart';
class CreateTodoWidget extends StatefulWidget {
 final TodoModel? todo;
 final TodoApiModel? todoApi;
 final ValueChanged<String> onSubmit;
 const CreateTodoWidget(
      {Key? key, this.todo, this.todoApi, required this.onSubmit})
      : super(key: key);
 State<CreateTodoWidget> createState() => _CreateTodoWidgetState();
class _CreateTodoWidgetState extends State<CreateTodoWidget> {
 final controller = TextEditingController();
 final formKey = GlobalKey<FormState>();
 void initState() {
   super.initState();
   controller.text = widget.todo?.title ?? '';
   controller.text = widget.todoApi?.title ?? '';
 @override
```

```
Widget build(BuildContext context) {
  final isUpdate = widget.todo != null || widget.todoApi != null;
  return AlertDialog(
    title: Text(isUpdate ? 'Edit todo' : 'Add Todo'),
    content: Form(
      key: formKey,
      child: TextFormField(
        controller: controller,
        autofocus: true,
        validator: (value) =>
            value != null && value.isEmpty ? 'Title is Required' : null,
    actions: [
      TextButton(
       onPressed: () {
          if (formKey.currentState!.validate()) {
            Navigator.pop(context);
        child: const Text("Cancel"),
      TextButton(
        onPressed: () {
          if (formKey.currentState!.validate()) {
            widget.onSubmit(controller.text);
        child: Text("Submit"),
   ],
```

todo_crud.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_1/meet89/create_todo_widget.dart';
import 'package:praktikum_1/meet89/model/todo_model.dart';
import 'package:praktikum_1/meet89/db/todo_db.dart';

// import 'DB/todo_db.dart';

// import 'package:praktikum_1/meet8/model/todo_model.dart';

class TodoCrud extends StatefulWidget {
   const TodoCrud({super.key});

   @override
   State<TodoCrud> createState() => _TodoCrudState();
}

class _TodoCrudState extends State<TodoCrud> {
   Future<List<TodoModel>>? futureTodo;
```

```
final todoDb = TodoDb();
void iniState() {
  super.initState();
 getTodo();
void getTodo() {
  setState(() {
    futureTodo = todoDb.getAll();
  });
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text("Todo API"),
    floatingActionButton: FloatingActionButton(
      onPressed: () {
        showDialog(
            context: context,
            builder: ( ) => CreateTodoWidget(onSubmit: (title) async {
                  await todoDb.create(title: title);
                  if (!mounted) return;
                  getTodo();
                  Navigator.of(context).pop();
                }));
      child: const Icon(Icons.add),
    body: FutureBuilder(
        future: futureTodo,
        builder: (context, snapshot) {
          if (snapshot.connectionState == ConnectionState.waiting) {
            return const Center(
              child: CircularProgressIndicator(),
          } else {
            if (snapshot.data == null) {
              return const Center(
                child: Text('No Todo Added Yet'),
            final todos = snapshot.data!;
            return todos.isEmpty
                ? const Center(
                    child: Text('No Todo Added yet'),
                : ListView.separated(
                    separatorBuilder: (context, Index) => const SizedBox(
                          height: 12,
                    itemBuilder: (context, index) {
                      final todo = todos[index];
```

```
return ListTile(
                title: Text(todo.title),
                trailing: IconButton(
                  onPressed: () async {
                     await todoDb.delete(todo.id);
                     getTodo();
                  icon: Icon(Icons.delete, color: Colors.pink),
                onTap: () {
                  showDialog(
                       context: context,
                       builder: (context) => CreateTodoWidget(
                           todo: todo,
                           onSubmit: (title) async {
                             await todoDb.update(
                                 id: todo.id, title: title);
                             getTodo();
                             if (!mounted) return;
                             Navigator.of(context).pop();
                           }));
            },
            itemCount: todos.length);
}),
```

- notifikasi.dart

```
import 'package:flutter_local_notifications/flutter_local_notifications.dart';
import 'package:praktikum_1/main.dart';

class Notifikasi {
    static FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin =
        FlutterLocalNotificationsPlugin();
    static Future
    NotificationResplugin();

static Future
    NotificationResponse notificationResponse) async {
    if (notificationResponse.payload != null) {
        String? payload = notificationResponse.payload;

    if (payload == 'second') {
        navKey.currentState?.pushNamed('/second');
      }
    }
}

static Future
    void> init() async {
    const AndroidInitializationSettings AndInitSet =
      AndroidInitializationSettings('@mipmap/ic_launcher');
    const DarwinInitializationSettings IOSinitSet =
```

```
DarwinInitializationSettings();
 const InitializationSettings InitSet =
     InitializationSettings(android: AndInitSet);
  await flutterLocalNotificationsPlugin.initialize(InitSet,
     onDidReceiveBackgroundNotificationResponse: onDidReceiveNotification,
      onDidReceiveNotificationResponse: onDidReceiveNotification);
  await flutterLocalNotificationsPlugin
      .resolvePlatformSpecificImplementation
          AndroidFlutterLocalNotificationsPlugin>()
      ?.requestNotificationsPermission();
static Future<void> showNotification(
    {required String title,
   required String body,
   required String payload}) async {
 const NotificationDetails platformChannelSpesifics = NotificationDetails(
      android: AndroidNotificationDetails('channelId', 'channelName',
          importance: Importance.max, priority: Priority.high));
  await flutterLocalNotificationsPlugin
      .show(0, title, body, platformChannelSpesifics, payload: payload);
```

- todo_api_model.dart

```
import 'dart:convert';

class TodoApiModel {
   String id;
   String title;

   TodoApiModel({required this.id, required this.title});

   factory TodoApiModel.fromJson(Map<String, dynamic> json) =>
        TodoApiModel(id: json["id"], title: json["title"]);

   Map<String, dynamic> toJson() => {"title": title, "id": id};
}

List<TodoApiModel> todoApiModelFromJson(String str) => List<TodoApiModel>.from(
   json.decode(str).map((x) => TodoApiModel.fromJson(x)));

String todoApiModelToJson(List<TodoApiModel> data) =>
   json.encode(List<dynamic>.from(data.map((x) => x.toJson())));
```

- api.dart

```
import 'package:http/http.dart';
import 'package:praktikum_1/API/model/todo_api_model.dart';
class TodoApi {
```

```
String baseUrl = "https://6767d3e0c1de2e6421c85934.mockapi.io/todo";
var client = Client();
Future<dynamic> get() async {
 var url = Uri.parse(baseUrl);
 var response = await client.get(url);
 if (response.statusCode == 200) {
   return todoApiModelFromJson(response.body);
  } else {
Future<dynamic> getById({required String id}) async {
 var url = Uri.parse("$baseUrl/$id");
 var response = await client.get(url);
 if (response.statusCode == 200) {
   return todoApiModelFromJson(response.body);
  } else {
Future<dynamic> post({required String title}) async {
 var url = Uri.parse(baseUrl);
 var body = {"title": title};
 var response = await client.post(url, body: body);
 if (response.statusCode == 201) {
   return response.body;
 } else {
Future<dynamic> put({required String id, required String title}) async {
 var url = Uri.parse("$baseUrl/$id");
 var body = {"title": title};
 var response = await client.put(url, body: body);
 if (response.statusCode == 200) {
   return (response.body);
 } else {
Future<dynamic> delete({required String id}) async {
 var url = Uri.parse("$baseUrl/$id");
 var response = await client.delete(url);
 if (response.statusCode == 200) {
   return response.body;
 } else {
```

```
}
```

- todo_api_crud.dart

```
import 'package:flutter/material.dart';
import 'package:path/path.dart';
import 'package:praktikum_1/API/api.dart';
import 'package:praktikum_1/API/view_todo_Api.dart';
import 'package:praktikum_1/meet89/db/todo_db.dart';
class todoApiCrud extends StatefulWidget {
 const todoApiCrud({super.key});
  State<todoApiCrud> createState() => todoApiCrudState();
class todoApiCrudState extends State<todoApiCrud> {
  Future<dynamic>? futureTodo;
  final todoDb = TodoDb();
  final todoApi = TodoApi();
  void initState() {
    super.initState();
   getTodo();
  void getTodo() {
    setState(() {
      futureTodo = todoApi.get();
    });
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Todo API"),
      floatingActionButton: FloatingActionButton(
          onPressed: () {
            showDialog(
                context: context,
                builder: (_) => View_Todo_API(onSubmit: (title) async {
                      await todoApi.post(title: title);
                      if (!mounted) return;
                      getTodo();
                      Navigator.of(context).pop();
                    }));
          child: const Icon(Icons.add)),
      body: FutureBuilder(
          future: futureTodo,
```

```
builder: (context, snapshot) {
  if (snapshot.data == null) {
    return const Center(
      child: CircularProgressIndicator(),
  final todos = snapshot.data!;
  return todos.isEmpty
       ? const Center(
           child: Text('No Todo Added Yet'),
       : ListView.separated(
           separatorBuilder: (context, index) => const SizedBox(
            height: 12,
           itemBuilder: (context, index) {
            final todo = todos[index];
             return ListTile(
                 title: Text(todo.title),
                 trailing: IconButton(
                   onPressed: () async {
                     await todoApi.delete(id: todo.id);
                     getTodo();
                   icon: Icon(Icons.delete),
                   color: Colors.red,
                 onTap: () {
                   showDialog(
                       context: context,
                       builder: (context) => View_Todo_API(
                           todoApi: todo,
                           onSubmit: (title) async {
                             await todoApi.put(
                                 id: todo.id, title: title);
                             getTodo();
                             if (!mounted) return;
                             Navigator.of(context).pop();
                           }));
                 });
           itemCount: todos.length,
}),
```

- view_todo_api.dart

```
import 'package:flutter/foundation.dart';
import 'package:flutter/material.dart';
import 'package:praktikum_1/API/model/todo_api_model.dart';
import 'package:praktikum 1/meet89/model/todo model.dart';
```

```
class View_Todo_API extends StatefulWidget {
 final TodoModel? todo;
 final TodoApiModel? todoApi;
 final ValueChanged<String> onSubmit;
 const View_Todo_API(
      {Key? key, this.todo, this.todoApi, required this.onSubmit})
      : super(key: key);
 @override
 State<View_Todo_API> createState() => _View_Todo_APIState();
class _View_Todo_APIState extends State<View_Todo_API> {
 final controller = TextEditingController();
 final formKey = GlobalKey<FormState>();
 @override
 void initState() {
   super.initState();
   controller.text = widget.todo?.title ?? '';
   controller.text = widget.todoApi?.title ?? '';
 Widget build(BuildContext context) {
    final isUpdate = widget.todo!=null || widget.todoApi != null;
   return AlertDialog(
      title: Text(isUpdate ? 'Edit todo' : 'Add Todo'),
     content: Form(
       key: formKey,
       child: TextFormField(
          controller: controller,
          autofocus: true,
          validator: (value) =>
              value != null && value.isEmpty ? 'Title is Required' : null,
      actions: [
        TextButton(
          onPressed: () {
            if (formKey.currentState!.validate()) {
              Navigator.pop(context);
          child: Text("Cancel"),
        TextButton(
         onPressed: () {
```

```
if (formKey.currentState!.validate()) {
        widget.onSubmit(controller.text);
     }
     // widget.onSubmit()
     },
     child: Text("Submit"),
     )
     ],
    );
}
```