

## PANDUAN TUGAS 2

### Persiapan Kode Awal Proyek Media Player

#### Capaian

1. Mahasiswa mampu mempersiapkan kode awal project flutter media\_player untuk membuat aplikasi pemutar media sederhana.

#### Spesifikasi Hardware dan Software

Memiliki komponen perangkat keras dan perangkat lunak yang benar sangat penting untuk memastikan keberhasilan pelaksanaan tugas yang diuraikan dalam panduan ini. Konfigurasi perangkat keras dan perangkat lunak yang diperlukan untuk menyelesaikan tugas panduan ini adalah sebagai berikut:

##### A. Spesifikasi Minimum Hardware

1. Minimum RAM 4 GB, disarankan RAM 8 GB
2. Minimum 15 GB ruang disk yang tersedia (2 GB untuk Flutter SDK, 8 GB untuk Android Studio, 4 GB untuk AVD, dan 1 GB untuk proyek)
3. Resolusi layar minimum 1280 x 800
4. Emulator Mobile (Android/IOS)

##### B. Software

1. Flutter SDK (versi 3.13.0 atau yang lebih baru) dan Dart (versi 3.1.5 atau yang lebih baru)
2. Android Studio
3. Visual Studio Code

#### Sumber Daya

1. [Repositori Starter Code media\\_player](#)
2. [Drive Starter Code \(khusus folder lib\)](#)
3. [Supplementary File](#)
4. [Test File Starter Code](#)

#### Deskripsi Tugas

Mahasiswa dapat mempersiapkan kode awal pada proyek Flutter media\_player. Mahasiswa akan mengatur environment proyek, menambahkan dependency, asset, dan kode konfigurasi awal pada proyek media\_player (pengaturan tema aplikasi, konstanta string, dan fungsi umum untuk manipulasi data). Setelah menambahkan semua kode yang diperlukan dalam konfigurasi awal ini, mahasiswa akan menjalankan testing starter code untuk memastikan konfigurasinya benar dan lengkap.

#### Langkah Praktikum

1. Clone starter code proyek dari [Starter Code media\\_player](#) atau ikuti langkah-langkah manual di bawah ini.

2. Buka proyek flutter media\_player yang telah anda buat pada panduan sebelumnya menggunakan Visual Studio Code.

3. Tambahkan package google\_fonts, flutter\_svg, intl, video\_player, audioplayers, dan cached\_network\_image. Jalankan perintah berikut pada terminal.

```
flutter pub add google_fonts, flutter_svg, intl, video_player, audioplayers, cached_network_image
```

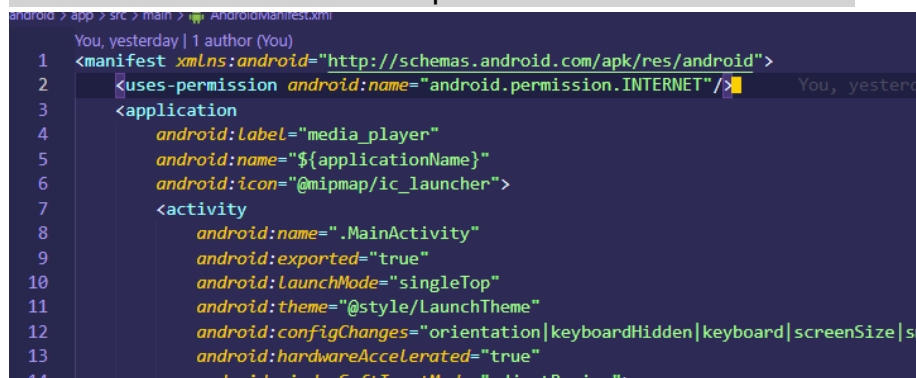
4. Tambahkan juga package mockito pada dev dependency dengan menjalankan perintah berikut pada terminal.

```
flutter pub add mockito --dev
```

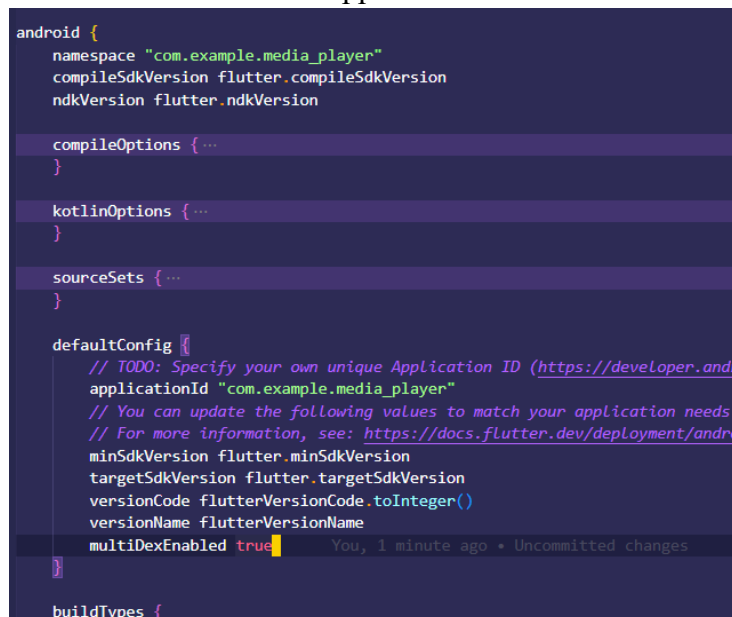
5. Tambahkan internet permission pada file AndroidManifest.xml yang terletak pada folder android>app>src>main.

```
<uses-permission
```

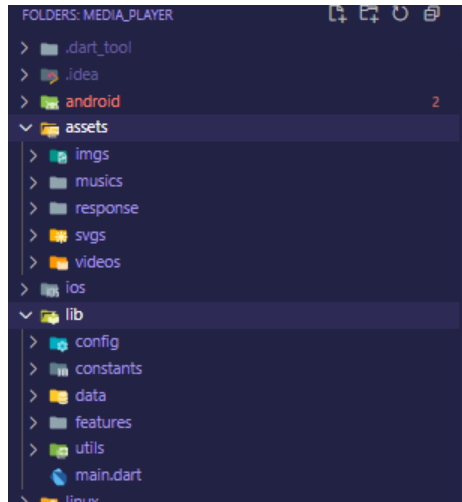
```
android:name="android.permission.INTERNET"/>
```



6. Tambahkan baris kode multiDexEnabled true pada file build.gradle yang terletak di dalam folder android>app.



7. Unduh starter code dari tautan berikut:
  - [Drive Starter Code \(khusus folder lib\)](#)
  - [Supplementary File](#)
8. Ekstrak dan masukan folder asset dan lib ke dalam root proyek.



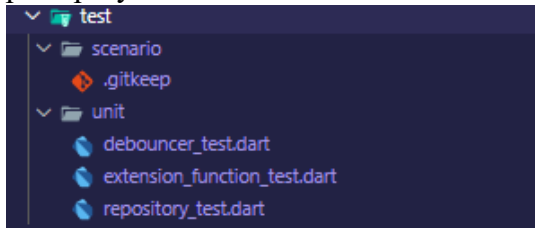
9. Daftarkan asset ke dalam file pubspec.yaml

```
pubspec.yaml
1 1  name: media_player
2 2  description: A new Flutter project.
3 3  publish_to: "none"
4 4
5 5  version: 1.0.0+1
6 6
7 7  environment:
8 8    sdk: ">=3.1.5 <4.0.0"
9 9  dependencies:
10 10   flutter:
11 11     sdk: flutter
12 12   cupertino_icons: ^1.0.2
13 13   google_fonts: ^6.2.0
14 14   flutter_svg: ^2.0.10+1
15 15   intl: ^0.19.0
16 16   video_player: ^2.8.3
17 17   audioplayers: ^5.2.1
18 18   cached_network_image: ^3.3.1
19 19
20 20  dev_dependencies:
21 21   flutter_test:
22 22     sdk: flutter
23 23   flutter_lints: ^2.0.0
24 24   mockito: ^5.4.4
25 25  flutter:
26 26    uses-material-design: true
27 27    assets:
28 28     - assets/svgs/
29 29     - assets/imgs/
30 30     - assets/musics/
31 31     - assets/videos/
32 32     - assets/response/
33 33
```

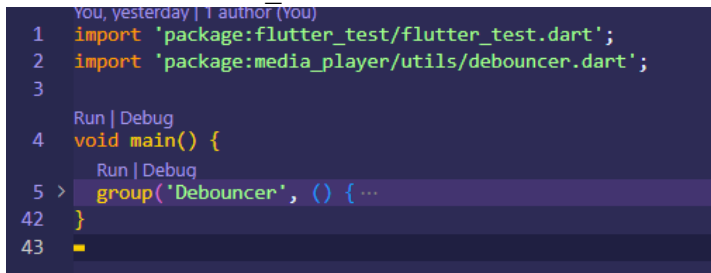
10. Pelajari setiap kode yang ada pada starter code tersebut, terutama kode yang ada pada folder lib dan assets.

## Langkah Verifikasi Kode

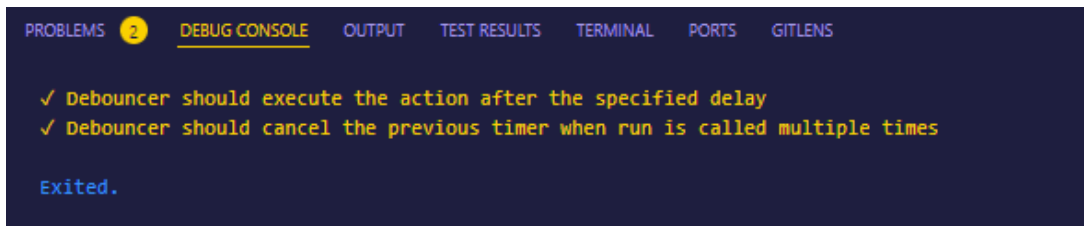
1. Unduh test file pada tautan [Test File Starter Code](#). Ekstrak dan timpa folder test yang ada pada proyek.



2. Buka file `debouncer_test.dart`.



3. Tekan tombol Run yang ada pada fungsi main dan tunggu proses hingga selesai.
4. Jika testing berhasil, maka akan muncul hasil berikut pada tab debug console Visual Studio Code anda.



\*Jika anda mengalami kegagalan, pastikan file `debouncer.dart` pada folder `lib>utils` pada proyek anda telah benar seperti pada tautan [berikut](#).

5. Lakukan hal yang sama untuk melakukan testing selanjutnya, yaitu untuk file **`extension_function_test.dart`** dan **`repository_test.dart`**.

\*Jika anda mengalami kegagalan, berikut referensi file yang bisa anda gunakan untuk memperbaiki kode anda.

- [extension\\_function.dart](#) (dalam folder `lib>utils`)
- [repository.dart](#) (dalam folder `lib>data`)

6. Anda juga bisa menjalankan perintah `flutter test` pada terminal untuk menjalankan semua test file yang ada pada folder test sekaligus. Jika starter code telah benar dan lengkap maka akan muncul hasil seperti berikut.

