

MODUL

PEMROGRAMAN BERBASIS PLATFORM



PERTEMUAN - 8

LIBRARY



PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ATMA JAYA YOGYAKARTA



2

DAFTAR GAMBAR

3	Gambar 1. Alur Penambahan Library	5
4	Gambar 2. Extension Pubspec Assist	5
5	Gambar 3. Library Responsive Sizer	5
6	Gambar 4 Tampilan tanpa menggunakan Library Responsive Sizer	6
7	Gambar 5 Tampilan dengan Menggunakan Library Responsive Sizer	6
8	Gambar 6 Library PDF	6
9	Gambar 7 Library Printing	7
10	Gambar 8 Library Image Picker	7
11	Gambar 9 Resiko Error pada Library Image Picker	8
12	Gambar 10 Library intl	8
13	Gambar 11 Library Barcode Widget.....	9
14	Gambar 12 Library uuid.....	9
15	Gambar 13 Menambahkan Library pada pubspec.yaml.....	10
16	Gambar 14 File main.dart.....	11
17	Gambar 15 Download Font	12
18	Gambar 16 Font di Folder Fonts	12
19	Gambar 17 Menambahkan Font pada pubspec.yaml.....	12
20	Gambar 18 Menambah Font pada Tema di File main.dart.....	13
21	Gambar 19 Isi Folder Model pada Folder Invoice.....	14
22	Gambar 20 Import yang ada di File home_page.dart	14
23	Gambar 21 stf Stateful Widget.....	14
24	Gambar 22 Deklarasi 1 HomePage	15
25	Gambar 23 Deklarasi 2 HomePage	15
26	Gambar 24 Future Image Picker	16
27	Gambar 25 Folder assets.....	16
28	Gambar 26 Pubspec Asset Logo	17
29	Gambar 27 Widget build Home Page.....	17
30	Gambar 28 App Bar Home Page.....	18
31	Gambar 29 Input Section Home Page	18
32	Gambar 30 Input Personal Data Bagian 1 Home Page	19
33	Gambar 31 Input Personal Data Bagian 2 Home Page	20
34	Gambar 32 Input Personal Data Bagian 2 Home Page	20
35	Gambar 33 Input Produk Home Page.....	21
36	Gambar 34 Pengaturan Button Create PDF Home Page	22
37	Gambar 35 Penanda Kurung Kurawal Buka Home Page State.....	23
38	Gambar 36 Product	23
39	Gambar 37 Custom Row	24



40	Gambar 38 Isi Folder Invoice	24
41	Gambar 39 File get_total_invoice.dart	25
42	Gambar 40 File item_doc.dart	26
43	Gambar 41 Preview Screen.....	27
44	Gambar 42 Permulaan Mengisi Kode pada Fungsi createPdf dalam File pdf_view.dart.....	28
46	Gambar 43 Parameter Fungsi createPdf	29
47	Gambar 44 Deklarasi dalam Fungsi createPdf.....	30
48	Gambar 45 Deklarasi untuk Memberi Value pada Tabel Invoice.....	31
49	Gambar 46 Menambah doc.addPage().....	31
50	Gambar 47 Menambah Navigator Setelah doc.addPage agar Tampil pada Preview Screen	32
52	Gambar 48 Isi Dalam Kurungnya doc.addPage.....	32
53	Gambar 49 Header PDF.....	33
54	Gambar 50 Image From Input	33
55	Gambar 51 Bagian 1 Personal Data Input	34
56	Gambar 52 Bagian 2 Personal Data Input	35
57	Gambar 53 Bagian Atas Invoice	36
58	Gambar 54 Bagian Isi Invoice	37
59	Gambar 55 Barcode Kotak.....	37
60	Gambar 56 Barcode Garis	38
61	Gambar 57 Footer PDF	38
62	Gambar 58 Hasil Jadi Bagian Atas Home Page	39
63	Gambar 59 Hasil Jadi Bagian Bawah Home Page	39
64	Gambar 60 Hasil Jadi Jika Input Gambar	40
65	Gambar 61 Hasil Jadi Jika Input Gambar dan Text Field	40
66	Gambar 62 Hasil Jadi Warning Jika Ada Text Field yang Belum Diisi.....	41
67	Gambar 63 Hasil Jadi Ketika Input Semuanya pada Home Page.....	41
68	Gambar 64 Hasil Jadi Preview dengan Tampilan Dokumen PDF Bagian Atas	42
69	Gambar 65 Hasil Jadi Preview dengan Tampilan Dokumen PDF Bagian Bawah	43
70	Gambar 66 Total Folder dan File yang Terbuat	44
71	Gambar 67 Hasil Tampilan Tabel Data Personal pada Dokumen PDF	45
72	Gambar 68 Kode di Dalam topOfInvoice untuk Membuat Informasi Berupa Tabel Tanpa Border	45
74	Gambar 69 Hasil Jadi Bagian Atas Invoice yang Berupa Tabel	45
75	Gambar 70 Hasil Jadi Tabel Invoice dengan Warna Row	46
76	Gambar 71 Deklarasi Value dan Tampilan dari File item_doc.dart di File pdf_view.dart dengan Value dari elements	46
78	Gambar 72 Pemanggilan table pada contentOfInvoice	46



79

80 TUJUAN

81

82 **Setelah menyelesaikan modul ini, praktikan diharapkan mampu :**

- 83 1. Mampu membuat tampilan yang responsif dengan library sizer
- 84 2. Mampu input gambar dan preview di halaman input ke dalam
- 85 dokumen PDF menggunakan Library image_picker
- 86 3. Mampu membuat dokumen, preview, print, share dokumen PDF
- 87 berdasarkan input menggunakan Library pdf dan printing pada
- 88 aplikasi yang dibuat.
- 89 4. Mampu memberi waktu dokumen PDF saat dibuat menggunakan
- 90 Library intl
- 91 5. Mampu memberi barcode di dalam dokumen PDF berisi kode unik
- 92 menggunakan Library barcode_widget dan uuid

93 DASAR TEORI

94

95 LIBRARY

96

97 Library merupakan sekumpulan kode yang biasanya terkumpul
98 dalam sebuah package yang memiliki fungsi-fungsi tertentu dan
99 dapat dipanggil ke dalam program lain untuk mempermudah
100 pembuatan dan pengembangan aplikasi. Pada saat penggeraan
101 proyek aplikasi, library ini dapat dipakai secara berulang-ulang.
102 Dengan library ini jika digunakan akan lebih mudah dibandingkan
103 menggunakan native code. Terdapat library-library yang dapat
104 digunakan. Berbagai library yang disediakan oleh Framework Flutter
105 dapat dilihat pada <https://pub.dev/>. Berikut library yang akan
106 dipelajari pada modul library ini.

107

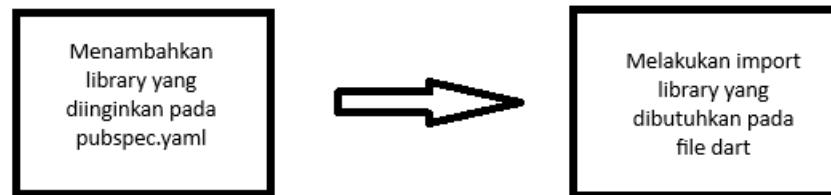
108

109



110

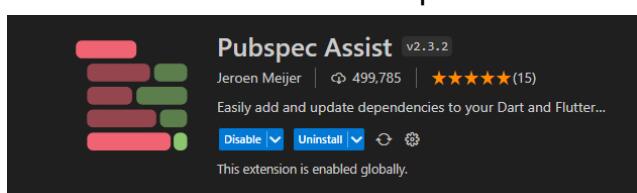
111 Alur penambahan Library:



112

113 Gambar 1. Alur Penambahan Library

114 Saat menambahkan library pada pubspec.yaml bisa menggunakan
115 extension bernama "Pubspec Assist"



116

117 Gambar 2. Extension Pubspec Assist

118

A. RESPONSIVE SIZER

119 Responsive Sizer adalah library yang digunakan untuk membuat
120 tampilan aplikasi menjadi lebih responsif yang dapat membuat
121 tampilan tetap bagus sesuai perangkat yang digunakan baik
122 smartphone, tablet, dekstop, maupun web dll.

123

124 Berikut ketentuan platform yang dapat menggunakan library Sizer:

125 responsive_sizer 3.2.0

Published 45 days ago • hanstan.link (Dart 3 compatible)

126

127

Gambar 3. Library Responsive Sizer

128 Perbandingan antara yang menggunakan responsive_sizer dengan
129 yang tidak:



Same UI in any Device
Responsiveness made Simple

Gambar 5 Tampilan dengan
Menggunakan Library Responsive Sizer



Gambar 4 Tampilan tanpa
menggunakan Library Responsive Sizer

130

131

132 Untuk keterangan lebih lanjutnya ada pada link ini:

https://pub.dev/packages/responsive_sizer

134

B. PDF

136 Library PDF merupakan library yang dapat membuat dokumen
137 PDF pada aplikasi. Dalam penggunaannya, library ini dapat
138 menghias, mengatur isi dokumennya, dan mengatur tata letaknya.
139 Platform yang dapat menggunakan library pdf adalah sebagai
140 berikut.

pdf 3.10.4

Published 3 months ago • nfet.net Dart 3 compatible

SDK DART FLUTTER PLATFORM ANDROID IOS LINUX MACOS WEB WINDOWS

141

142 Gambar 6 Library PDF

143

144 Untuk keterangan lebih lanjutnya ada pada link ini:

<https://pub.dev/packages/pdf>

146

C. PRINTING

148 Library Printing adalah library yang dapat mendukung library
149 pdf untuk print atau cetak dokumen pdfnya. Dengan library ini, maka



150 dokumen pdf dapat dipreview dan print di dalam aplikasi. Platform
151 yang dapat menggunakan library pdf adalah sebagai berikut.
152

printing 5.11.0

Published 4.months ago • nfet.net Dart 3 compatible

SDK FLUTTER PLATFORM ANDROID IOS LINUX MACOS WEB WINDOWS

153

154 Gambar 7 Library Printing

155 Untuk keterangan lebih lanjutnya ada pada link ini:

156 <https://pub.dev/packages/printing>

157

D. IMAGE PICKER

159 Library Image Picker adalah library untuk memilih gambar dari
160 galeri dan kamera. Namun, pada library ini tidak support pada web.
161 Maka dari itu, jika ingin menggunakan pada web, perlu untuk
162 menggunakan **image_picker_for_web**.

163 Platform yang dapat menggunakan library pdf adalah sebagai
164 berikut.

image_picker 1.0.4

Published 51.days.ago • flutter.dev Dart 3 compatible

SDK FLUTTER PLATFORM ANDROID IOS LINUX MACOS WEB WINDOWS

5.9K

Readme Changelog Example Installing Versions Scores

Image Picker plugin for Flutter

pub v1.0.4

A Flutter plugin for iOS and Android for picking images from the image library, and taking new pictures with the camera.

Android iOS Linux macOS Web

Windows

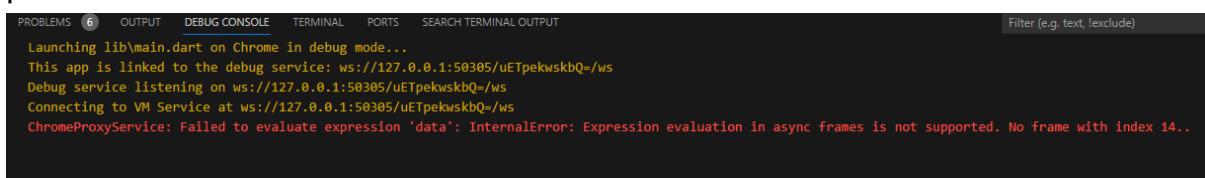
Support SDK 21+ iOS 11+ Any 10.14+ See [image_picker_for_web](#) Windows 10+

165

166 Gambar 8 Library Image Picker



167 Jika pada web akan muncul error seperti ini saat klik button create
168 pdf:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH TERMINAL OUTPUT
Launching lib/main.dart on Chrome in debug mode...
This app is linked to the debug service: ws://127.0.0.1:50305/uETpekwskbQ=/ws
Debug service listening on ws://127.0.0.1:50305/uETpekwskbQ=/ws
Connecting to VM Service at ws://127.0.0.1:50305/uETpekwskbQ=/ws
ChromeProxyService: Failed to evaluate expression 'data': InternalError: Expression evaluation in async frames is not supported. No frame with index 14..
```

169

170 Gambar 9 Resiko Error pada Library Image Picker

171 "Expression evaluation in async frames is not supported"

172 Sehingga untuk **guided modul ini jangan dijalankan pada web**. Selain
173 web akan aman untuk dijalankan saat klik button create pdf.

174

175 Untuk keterangan lebih lanjutnya ada pada link ini:

176 https://pub.dev/packages/image_picker

177

E. INTL

179 Library INTL adalah library yang dapat mendefinisikan
180 DateFormat, NumberFormat, dan BidiFormat. Untuk format tersebut,
181 dapat berasal dari lokal maupun internasional. Platform yang dapat
182 menggunakan library pdf adalah sebagai berikut.

intl 0.18.1

Published 6.months.ago • ⚡ dart.dev (Dart 3 compatible)

183 [SDK](#) [DART](#) [FLUTTER](#) [PLATFORM](#) [ANDROID](#) [IOS](#) [LINUX](#) [MACOS](#) [WEB](#) [WINDOWS](#)

184

Gambar 10 Library intl

185 Untuk keterangan lebih lanjutnya ada pada link ini:

186 <https://pub.dev/packages/intl>

187

F. BARCODE_WIDGET

189 Library Barcode Widget adalah library untuk membuat atau
190 generate barcode berisi sebuah data. Barcode ini dapat berupa



191 beberapa macam. Platform yang dapat menggunakan library pdf
192 adalah sebagai berikut.

barcode_widget 2.0.4

Published 5 months ago • nfet.net (Dart 3 compatible)

SDK FLUTTER PLATFORM ANDROID IOS LINUX MACOS WEB WINDOWS

193

194 Gambar 11 Library Barcode Widget

195 Untuk keterangan lebih lanjutnya ada pada link ini:
196 https://pub.dev/packages/barcode_widget

G. UUID

198 Library UUID adalah library yang digunakan untuk membuat
199 nomor acak dari suatu data. UUID terdiri dari beberapa macam
200 antara lain: RFC4122 (v1, v4, v5, v6, v7, v8). Platform yang dapat
201 menggunakan library pdf adalah sebagai berikut.

uuid 4.1.0

Published 25 days ago • yuli.dev (Dart 3 compatible)

SDK DART FLUTTER PLATFORM ANDROID IOS LINUX MACOS WEB WINDOWS

202

203 Gambar 12 Library uuid

204 Untuk keterangan lebih lanjutnya ada pada link ini:
205 <https://pub.dev/packages/uuid>

206

207

208

209



GUIDED 8– Library

212 **Poin yang dipelajari dalam Guided 8 ini, yaitu :**

- 213 1. Memahami cara untuk menerapkan library Sizer untuk membuat
214 aplikasi dengan tampilan responsif pada proyek Flutter
- 215 2. Memahami cara untuk menerapkan beberapa library untuk
216 membuat dokumen pdf pada proyek Flutter
- 217 3. Memahami macam-macam library yang ada di Flutter

218 Pada guided ini, kita akan mulai untuk membuat aplikasi yang dapat
219 membuat suatu dokumen pdf berdasarkan input dan menghias dokumen
220 pdf nya, :

- 221 1. Buka aplikasi Visual Studio Code, tekan CTRL+SHIFT+P (Command
222 Pallete) pilih command Flutter : new project. Kemudian, pilih **Empty
223 Application**
- 224 2. Pilih lokasi penyimpanan untuk menyimpan projek guidednya, Isi
225 nama file "**gdlibrary_kelas_xxxx**"(xxxx: 4 NPM terakhir).
- 226 3. Pada **pubspec.yaml**, tambahkan ini untuk menggunakan library-
227 librarynya

```
dependencies:  
  flutter:  
    sdk: flutter  
  
  cupertino_icons: ^1.0.2  
  
  pdf: ^3.6.4  
  printing: ^5.9.0  
  image_picker: ^1.0.4  
  intl: ^0.17.0  
  barcode_widget: ^2.0.4  
  uuid: ^3.0.6  
  responsive_sizer: ^3.2.0
```

228

229 Gambar 13 Menambahkan Library pada pubspec.yaml

230 Dikarenakan saat membuat projek menggunakan empty application,
231 maka harus menambahkan library **cupertino_icons** secara manual.

232

233 **4.** Kemudian, buat main.dart nya menjadi seperti ini



```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:gdlibrary_kelas_xxxx/home_page.dart';
import 'package:responsive_sizer/responsive_sizer.dart';

void main() {
    runApp(const MyApp());
}

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

    // This widget is the root of your application.
    @override
    Widget build(BuildContext context) {
        return ResponsiveSizer(builder: (context, orientation, deviceType) {
            Device.orientation == Orientation.portrait
                ? Container( // Widget for Portrait
                    width: 100.w,
                    height: 20.5.h,
                )
                : Container( // Widget for Landscape
                    width: 100.w,
                    height: 12.5.h,
                );
            Device.screenType == ScreenType.tablet
                ? Container( // Widget for Tablet
                    width: 100.w,
                    height: 20.5.h,
                )
                : Container( // Widget for Mobile
                    width: 100.w,
                    height: 12.5.h,
                );
        );
        return MaterialApp(
            title: 'Flutter Demo',
            debugShowCheckedModeBanner: false,
            theme: ThemeData(
                colorScheme: ColorScheme.fromSeed(seedColor: Colors.amber)
                    .copyWith(
                        secondary: Colors.amberAccent,
                    ),
                materialTapTargetSize: MaterialTapTargetSize.shrinkWrap,
                appBarTheme: const AppBarTheme(
                    systemOverlayStyle: SystemUiOverlayStyle.dark,
                ),
                useMaterial3: true,
            ),
            home: const MyHomePage(),
        );
    }
}
```

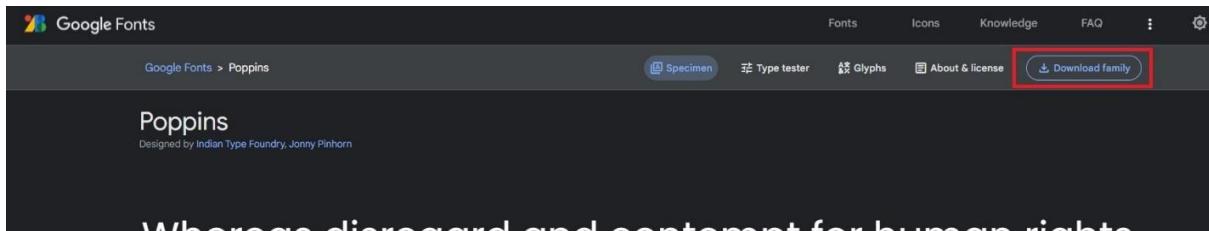
234

235

Gambar 14 File main.dart



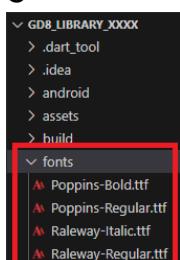
- 236 5. Download font pada link berikut <https://fonts.google.com/>, cari font
237 **raleway** dan **poppins**



238 Whereas disregard and contempt for human rights

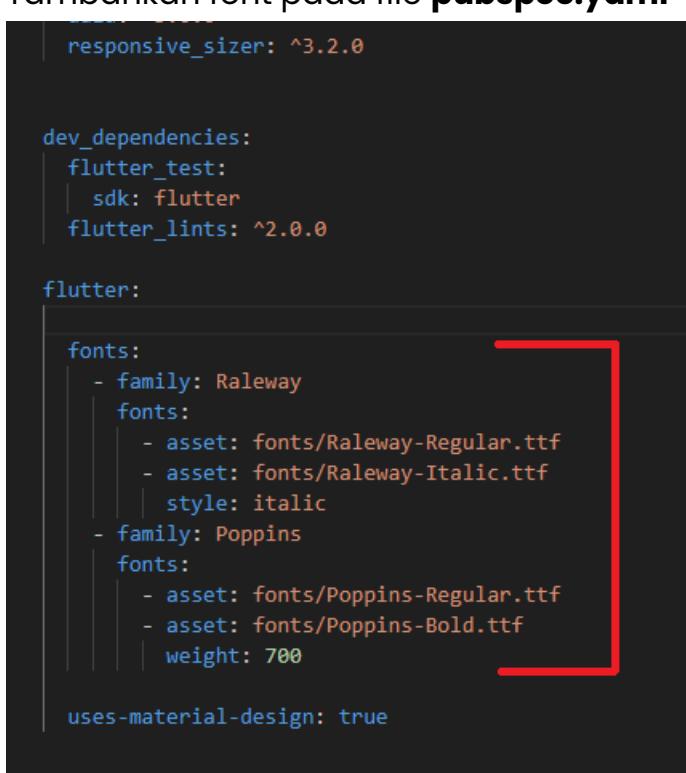
239 Gambar 15 Download Font

- 240
241 6. Tambah folder **fonts** pada projek, pilih file font sesuai yang ada pada
242 gambar dari hasil download font dengan copy paste ke folder **fonts**



243
244 Gambar 16 Font di Folder Fonts

- 245 7. Tambahkan font pada file **pubspec.yaml**



246
247 Gambar 17 Menambahkan Font pada pubspec.yaml



248

8. Kemudian, tambahkan font untuk tema aplikasi

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:gdlibrary_kelas_xxxx/home_page.dart';
import 'package:responsive_sizer/responsive_sizer.dart';

void main() {
    runApp(const MyApp());
}

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

    // This widget is the root of your application.
    @override
    Widget build(BuildContext context) {
        return ResponsiveSizer(builder: (context, orientation, deviceType) {
            {
                Device.orientation == Orientation.portrait
                    ? Container( // Widget for Portrait
                        width: 100.w,
                        height: 20.5.h,
                    )
                    : Container( // Widget for Landscape
                        width: 100.w,
                        height: 12.5.h,
                    );
                Device.screenType == ScreenType.tablet
                    ? Container( // Widget for Tablet
                        width: 100.w,
                        height: 20.5.h,
                    )
                    : Container( // Widget for Mobile
                        width: 100.w,
                        height: 12.5.h,
                    );
            );
            return MaterialApp(
                title: 'Flutter Demo',
                debugShowCheckedModeBanner: false,
                theme: ThemeData(
                    fontFamily: 'Poppins', ————— tambahkan 1 baris kode ini
                    colorScheme: ColorScheme.fromSeed(seedColor: Colors.amber)
                        .copyWith(
                            secondary: Colors.amberAccent,
                        ),
                    materialTapTargetSize: MaterialTapTargetSize.shrinkWrap,
                    appBarTheme: const AppBarTheme(
                        systemOverlayStyle: SystemUiOverlayStyle.dark,
                    ),
                    useMaterial3: true,
                ),
                home: const MyHomePage(),
            );
        },);
    }
}
```

249

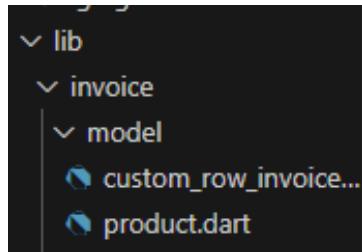
250

Gambar 18 Menambah Font pada Tema di File main.dart

251

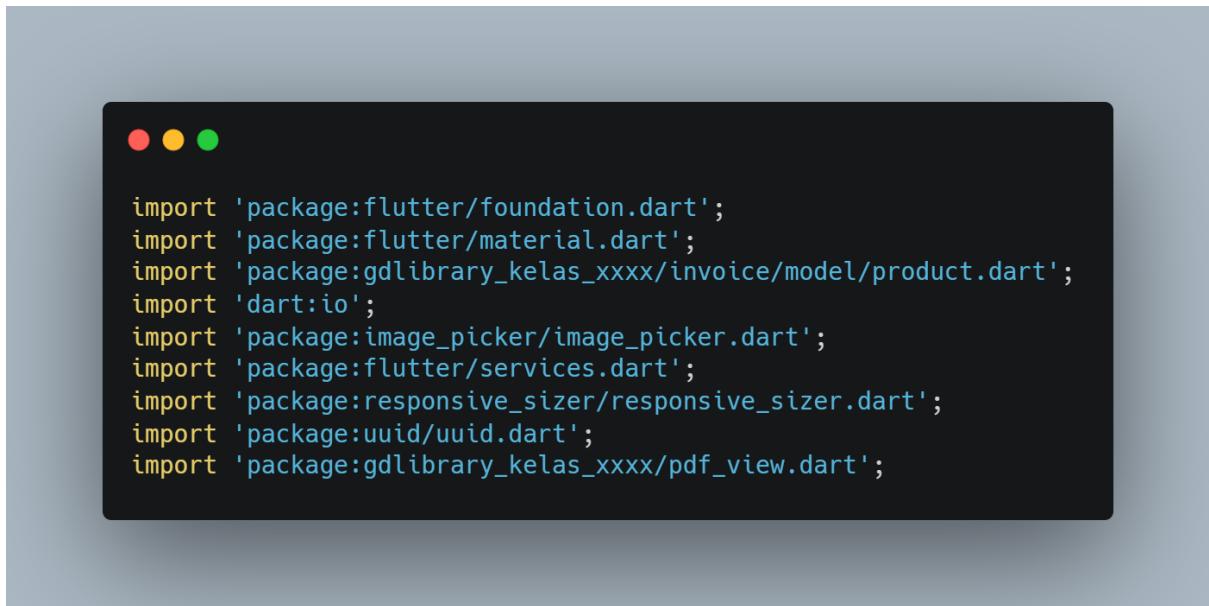


252 9. Selanjutnya, buat folder **invoice**, di dalam folder invoice buatlah folder
253 **model**, di dalam folder model buatlah 2 file dengan nama
254 **custom_row_invoice.dart** dan **product.dart** menjadi seperti ini:



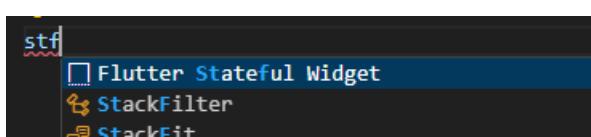
255
256 Gambar 19 Isi Folder Model pada Folder Invoice

257 10. Selanjutnya, buat file **home_page.dart** yang akan digunakan
258 sebagai halaman input di dalam folder **lib** dan import hal-hal yang
259 dibutuhkan, antara lain:



260
261 Gambar 20 Import yang ada di File home_page.dart

262
263 11. Ketik **stf** kemudian klik Flutter Stateful Widget



264
265 Gambar 21 stf Stateful Widget
266 12. Kemudian, isilah untuk mendeklarasikan
267 -controller,



268 -file untuk gambar,
269 -kode unik(menggunakan library uuid),
270 -value produk untuk tampilan invoice,
271 Seperti pada gambar berikut.

```
class MyHomePage extends StatefulWidget {  
    const MyHomePage({Key? key}) : super(key: key);  
  
    @override  
    State<MyHomePage> createState() => _MyHomePageState();  
}  
  
class _MyHomePageState extends State<MyHomePage> {  
    final nameController = TextEditingController();  
    final phoneController = TextEditingController();  
    final addressController = TextEditingController();  
    String id = const Uuid().v1(); // untuk membuat kode unik berdasarkan waktu  
    File? image;
```

272

273

Gambar 22 Deklarasi 1 HomePage

274

```
//List untuk value dari produknya  
List<Product> products = [  
    Product("Membership", 50000),  
    Product("Books", 30000),  
    Product("Pen", 5000),  
    Product("Snacks", 10000),  
    Product("Hamburger", 25000),  
];  
int number = 0;  
getTotal() => products  
    .fold(0.0,  
        (double prev, element) => prev + (element.price * element.amount))  
    .toStringAsFixed(2);  
  
getPPN() => products  
    .fold(  
        0.0,  
        (double prev, element) =>  
            prev +  
                (element.price / 100 * element.ppnInPercent * element.amount))  
    .toStringAsFixed(2);
```

275

276

Gambar 23 Deklarasi 2 HomePage

277

Deklarasi ini akan berguna pada halaman input.

278

279 13. Memanfaatkan proses asinkron untuk input gambar dari gallery dan
280 camera menggunakan future. Di dalam fungsinya menggunakan
281 library **image_picker**.

```
// untuk input pemilihan gambar dari galeri
Future pickImage() async {
  try {
    final image = await ImagePicker().pickImage(source: ImageSource.gallery);

    if (image == null) return;

    final imageTemp = File(image.path);
    setState(() => this.image = imageTemp);
  } on PlatformException catch (e) {
    debugPrint(
      'Failed to pick image: $e'); //saran untuk ke depannya: cek saat di log dan
sebaiknya dihapus ketika deploy app, namun dapat ditambah dengan toast atau snackbars
  }
}

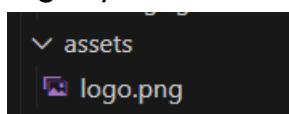
//untuk input pengambilan gambar dari kamera
Future pickImageC() async {
  try {
    final image = await ImagePicker().pickImage(source: ImageSource.camera);

    if (image == null) return;
    final imageTemp = File(image.path);
    setState(() => this.image = imageTemp);
  } on PlatformException catch (e) {
    debugPrint(
      'Failed to pick image: $e'); //saran untuk ke depannya: cek saat di log dan
sebaiknya dihapus ketika deploy app, namun dapat ditambah dengan toast atau snackbars
  }
}
```

282

Gambar 24 Future Image Picker

284 14. Buat folder asset di dalam file projek guided, kemudian isi dengan
285 gambar untuk logo invoicenya, terserah pilih yang mana untuk
286 logonya, bisa dicari internet, menjadi seperti ini:



287

Gambar 25 Folder assets

289

290

291



292 15. Tambahkan kode yang di kotak merah agar asset dapat digunakan

```
  sdk: flutter
  flutter_lints: ^2.0.0

  flutter:
    assets:
      - assets/logo.png

    fonts:
      - family: Raleway
        fonts:
          - asset: fonts/Raleway-Regular.ttf
          - asset: fonts/Raleway-Italic.ttf
            style: italic
      - family: Poppins
        fonts:
          - asset: fonts/Poppins-Regular.ttf
          - asset: fonts/Poppins-Bold.ttf
            weight: 700
```

293

294 Gambar 26 Pubspec Asset Logo

295 16. Mulai pembuatan tampilan halaman inputnya yang terdiri dari

296 **appBar** dan **inputSection** nya



297

298 Gambar 27 Widget build Home Page

299 Note: **Isi kode** dari AppBar dan Input Section ada **di dalam kelas**
300 **MyHomePageState**, namun **di luar Widget build**.

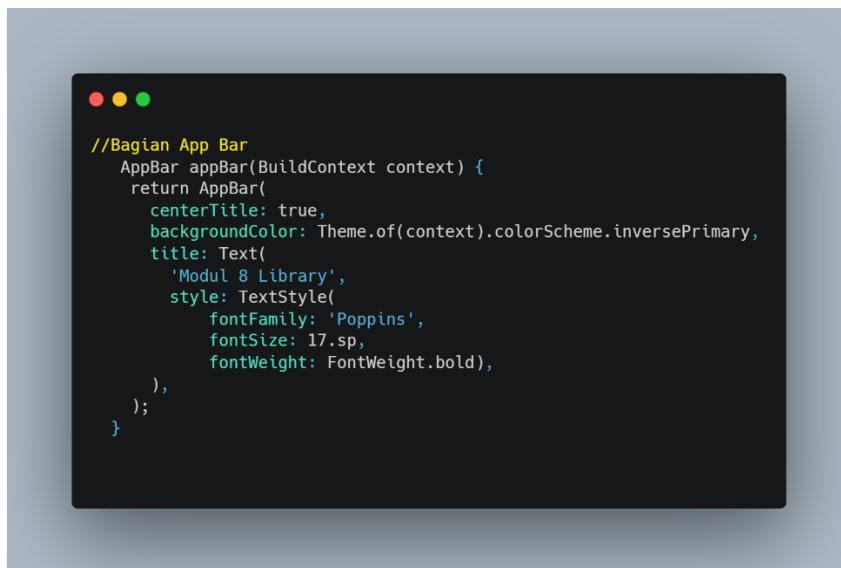
301

302



303

AppBarnya berisi ini:



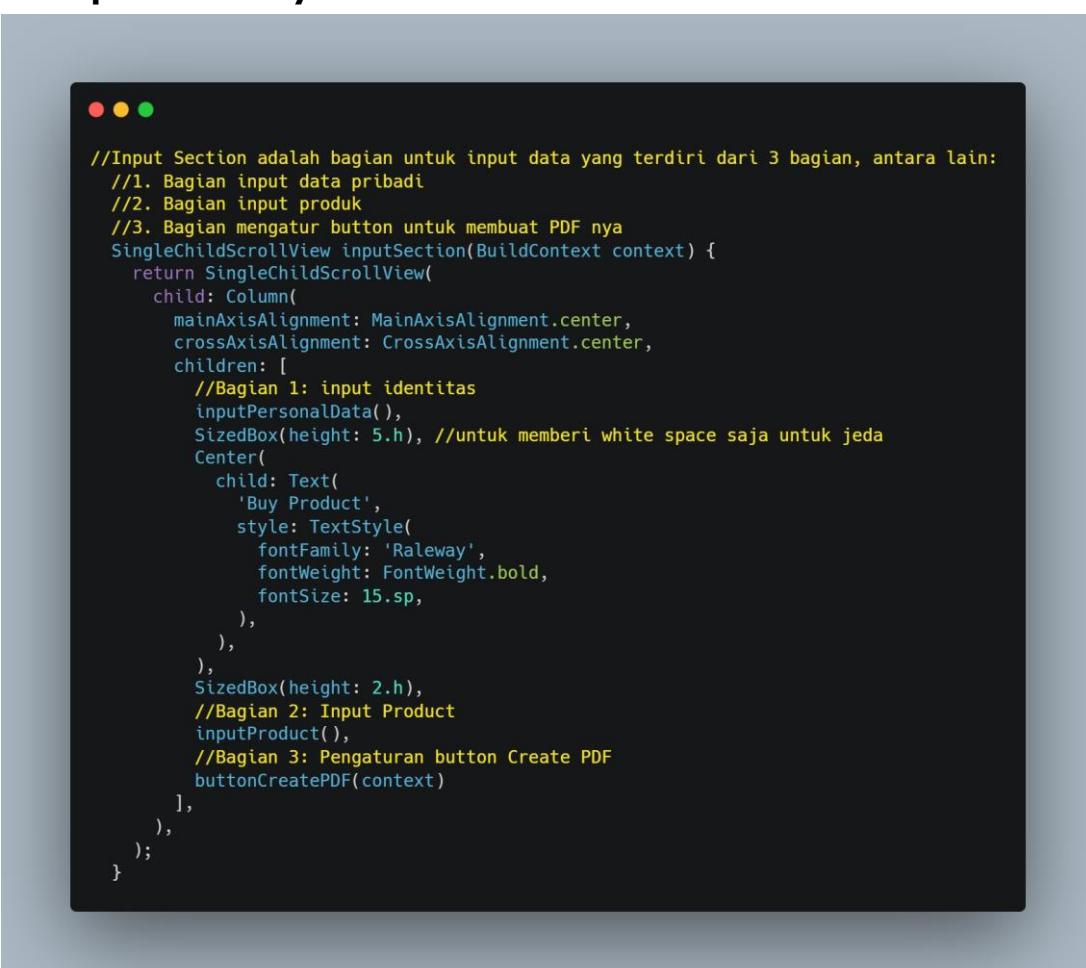
304

305

Gambar 28 App Bar Home Page

306

InputSectionnya berisi ini:



307

308

Gambar 29 Input Section Home Page



309 Note: **Isi Kode Bagian Input Personal Data, Input Produk, dan**
310 **Pengaturan Button Create PDF berada di luar inputSection, namun**
311 **masih di dalam kelas MyHomePageState**

312

313 **Bagian 1** dari inputSection adalah **inputPersonalData**:



```
//1. Bagian input data pribadi
Container inputPersonalData() {
    return Container(
        margin: EdgeInsets.all(3.h),
        child: Column(children: [
            Center(
                child: Text(
                    'Fill This Form',
                    style: TextStyle(
                        fontFamily: 'Raleway',
                        fontWeight: FontWeight.bold,
                        fontSize: 15.sp,
                    ),
                ),
            ),
            Column(
                mainAxisAlignment: MainAxisAlignment
                    .center, // Center the buttons horizontally
                children: [
                    Padding(
                        padding:
                            EdgeInsets.symmetric(horizontal: 2.h, vertical: 2.h),
                        child: ElevatedButton(
                            style: ElevatedButton.styleFrom(
                                backgroundColor: Colors.amber,
                                textStyle: TextStyle(
                                    fontWeight: FontWeight.bold,
                                    fontSize: 15.sp,
                                ),
                            ),
                            onPressed: () {
                                pickImageC();
                            },
                            child: const Text("Pick Image from Camera"),
                        ),
                    ),
                    Padding(
                        padding:
                            EdgeInsets.symmetric(horizontal: 2.h, vertical: 2.h),
                        child: ElevatedButton(
                            style: ElevatedButton.styleFrom(
                                padding: EdgeInsets.symmetric(horizontal: 2.h),
                                backgroundColor: Colors.amber,
                                textStyle: TextStyle(
                                    fontWeight: FontWeight.bold,
                                    fontSize: 15.sp,
                                ),
                            ),
                            onPressed: () {
                                pickImage();
                            },
                            child: const Text("Pick Image from Gallery"),
                        ),
                    ),
                    SizedBox(
                        width: 10.w), // Add some spacing between the buttons
                ],
            ),
        ],
    );
}
```

314

315

Gambar 30 Input Personal Data Bagian 1 Home Page



316

```
SizedBox(  
    height: 2.h,  
,  
    if (image != null)  
        Column(  
            children: [  
                //jika pada web menggunakan Image.network, jika selain web  
menggunakan Image.file  
                kIsWeb ? Image.network(image!.path) : Image.file(image!),  
                Padding(  
                    padding: EdgeInsets.symmetric(horizontal: 2.h),  
                    child: IconButton(  
                        color: Colors.red,  
                        iconSize: 5.h,  
                        icon: const Icon(Icons.delete),  
                        onPressed: () {  
                            setState(() {  
                                image = null;  
                            });  
                        },  
                    ),  
                ),  
            ],  
        )  
    else  
        const Text("No image selected"),  
    TextFormField(  
        controller: nameController,  
        decoration: const InputDecoration(  
            hintText: 'Enter Your Name', labelText: 'Name'),  
,  
),
```

317

Gambar 31 Input Personal Data Bagian 2 Home Page

318

```
TextFormField(  
    controller: phoneController,  
    decoration: const InputDecoration(  
        hintText: 'Enter Your Phone Number',  
        labelText: 'Phone Number'),  
    keyboardType: TextInputType.number,  
,  
    TextFormField(  
        controller: addressController,  
        decoration: const InputDecoration(  
            hintText: 'Enter Your Address', labelText: 'Address'),  
        ),  
    ],  
,  
);  
}
```

319

Gambar 32 Input Personal Data Bagian 2 Home Page

321

322

323

324

325



326

Bagian 2 dari inputSection adalah **input produk**

```
//2. Bagian input produk
Padding inputProduct() {
  return Padding(
    padding: EdgeInsets.symmetric(horizontal: 2.h),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.center,
      children: [
        SizedBox(
          height: 180,
          child: ListView.builder(
            shrinkWrap: true,
            itemBuilder: (context, index) {
              final currentProduct = products[index];
              return Row(
                children: [
                  Expanded(child: Text(currentProduct.name)),
                  Expanded(
                    child:
                      Text("Price: Rp ${currentProduct.price.toStringAsFixed(2)}")),
                  Expanded(
                    child: Row(
                      children: [
                        Expanded(
                          child: IconButton(
                            onPressed: () {
                              setState(() => currentProduct.amount++);
                            },
                            icon: const Icon(Icons.add),
                          ),
                        ),
                        Expanded(
                          child: Text(
                            currentProduct.amount.toString(),
                            textAlign: TextAlign.center,
                          ),
                        ),
                        Expanded(
                          child: IconButton(
                            onPressed: () {
                              setState(() => currentProduct.amount--);
                            },
                            icon: const Icon(Icons.remove),
                          ),
                        ),
                      ],
                    ),
                  ),
                ],
              );
            },
            itemCount: products.length,
          ),
        ),
        Row(
          mainAxisAlignment: MainAxisAlignment.spaceBetween,
          children: [const Text("PPN(11%)"), Text("Rp ${getPPN()}")],
        ),
        Row(
          mainAxisAlignment: MainAxisAlignment.spaceBetween,
          children: [const Text("Total"), Text("Rp ${getTotal()}")],
        ),
      ],
    );
}
```

327

328

Gambar 33 Input Produk Home Page



329

330

331 **Bagian 3** dari inputSection adalah Bagian **mengatur button** untuk
332 membuat PDF nya, sebagai berikut.

The screenshot shows a dark-themed mobile application interface. At the top, there are three circular icons: red, yellow, and green. Below them is a code editor window displaying the following Dart code:

```
//3. Bagian mengatur button untuk membuat PDF nya
Container buttonCreatePDF(BuildContext context) {
    return Container(
        margin: EdgeInsets.symmetric(vertical: 2.h),
        child: ElevatedButton(
            onPressed: () {
                if (image == null ||
                    nameController.text.isEmpty ||
                    phoneController.text.isEmpty ||
                    addressController.text.isEmpty) {
                    showDialog(
                        context: context,
                        builder: (context) => AlertDialog(
                            title: const Text('Warning'),
                            content: const Text(
                                'Please fill in all the fields.'),
                            actions: [
                                TextButton(
                                    onPressed: () => Navigator.pop(context),
                                    child: const Text('OK'),
                                ),
                            ],
                        );
                }
                return; // Stop PDF creation if fields are empty
            } else {
                createPdf(nameController,
                    phoneController, addressController, id, image!, context, products);
                setState(() {
                    const uuid = Uuid();
                    id = uuid.v1();
                });
            },
            style: ElevatedButton.styleFrom(
                backgroundColor: Colors.amber,
                textStyle: TextStyle(
                    fontWeight: FontWeight.bold,
                    fontSize: 15.sp,
                ),
                child: const Text('Create PDF'),
            ),
        );
    );
}
```

333

334

Gambar 34 Pengaturan Button Create PDF Home Page



335
336 Note: Jangan lupa 1 kurung kurawal tutup sesudah bagian 3 yang
337 merupakan sambungan kurung kurawal buka dari
338 `class _MyHomePageState extends State<MyHomePage> {
| final nameController = TextEditingController();`

339 Gambar 35 Penanda Kurung Kurawal Buka Home Page State

340
341 17. Langkah berikutnya adalah mengisi file **product** yang ada di model
342 sebagai berikut.



343
344 Gambar 36 Product
345 18. Kemudian, isi kode pada file **custom_row_invoice**, sebagai berikut.

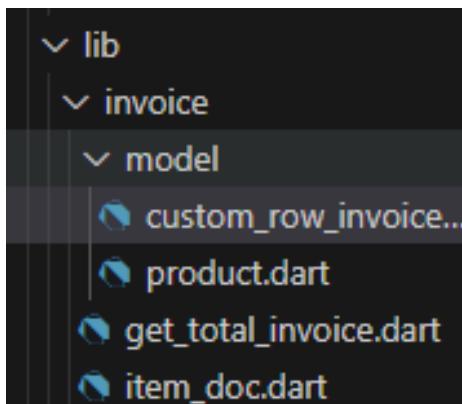


```
class CustomRow {  
    final String itemName;  
    final String itemPrice;  
    final String amount;  
    final String subTotalProduct;  
  
    CustomRow(  
        this.itemName,  
        this.itemPrice,  
        this.amount,  
        this.subTotalProduct);  
}
```

346

347 Gambar 37 Custom Row

348 19. Setelah itu, buat file dengan nama "**get_total_invoice**" dan
349 "**item_doc**" di dalam folder **invoice**



350

351 Gambar 38 Isi Folder Invoice

352

353 20. Isilah kode pada file **get_total_invoice**, sebagai berikut:



```
import 'package:gdlibrary_kelas_xxxx/invoice/model/product.dart';

String getSubTotal(List<Product> products) {
    return products
        .fold(0.0,
            (double prev, element) => prev + (element.amount * element.price))
        .toStringAsFixed(2);
}

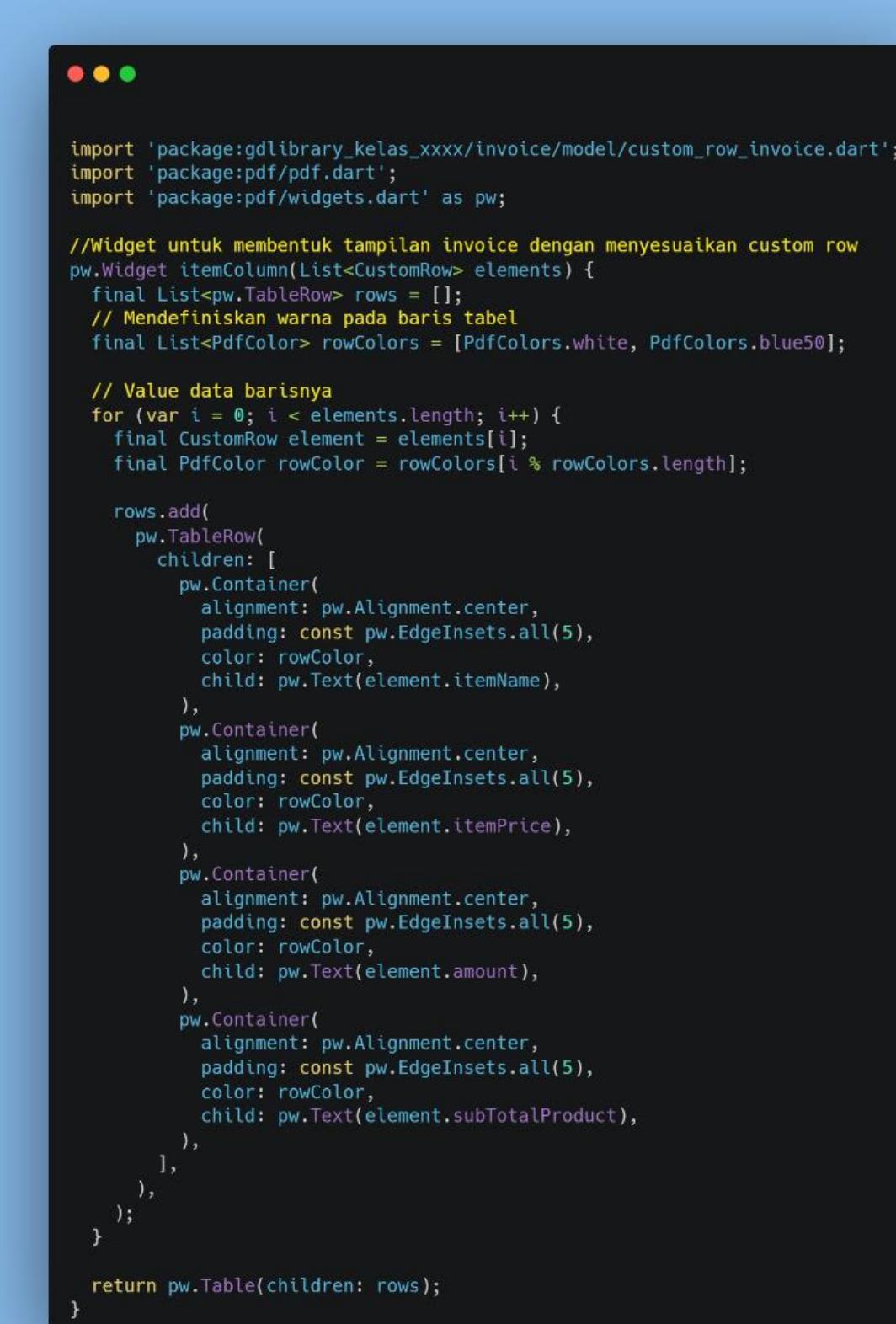
String getPPNTotal(List<Product> products) {
    return products
        .fold(
            0.0,
            (double prev, next) =>
                prev + ((next.price / 100 * next.ppnInPercent) * next.amount),
        )
        .toStringAsFixed(2);
}
```

354

355 Gambar 39 File get_total_invoice.dart

356

357 21. Isilah juga kode di dalam file **item_doc.dart**, sebagai berikut.



```
import 'package:gdlibrary_kelas_xxxx/invoice/model/custom_row_invoice.dart';
import 'package:pdf/pdf.dart';
import 'package:pdf/widgets.dart' as pw;

//Widget untuk membentuk tampilan invoice dengan menyesuaikan custom row
pw.Widget itemColumn(List<CustomRow> elements) {
    final List<pw.TableRow> rows = [];
    // Mendefinisikan warna pada baris tabel
    final List<PdfColor> rowColors = [PdfColors.white, PdfColors.blue50];

    // Value data barisnya
    for (var i = 0; i < elements.length; i++) {
        final CustomRow element = elements[i];
        final PdfColor rowColor = rowColors[i % rowColors.length];

        rows.add(
            pw.TableRow(
                children: [
                    pw.Container(
                        alignment: pw.Alignment.center,
                        padding: const pw.EdgeInsets.all(5),
                        color: rowColor,
                        child: pw.Text(element.itemName),
                    ),
                    pw.Container(
                        alignment: pw.Alignment.center,
                        padding: const pw.EdgeInsets.all(5),
                        color: rowColor,
                        child: pw.Text(element.itemPrice),
                    ),
                    pw.Container(
                        alignment: pw.Alignment.center,
                        padding: const pw.EdgeInsets.all(5),
                        color: rowColor,
                        child: pw.Text(element.amount),
                    ),
                    pw.Container(
                        alignment: pw.Alignment.center,
                        padding: const pw.EdgeInsets.all(5),
                        color: rowColor,
                        child: pw.Text(element.subTotalProduct),
                    ),
                ],
            ),
        );
    }

    return pw.Table(children: rows);
}
```

358

359

Gambar 40 File item_doc.dart

360



361
362
363
364
365
366

22. Selanjutnya buat file **preview_screen.dart** di dalam folder **lib** yang berisi kode ini:

```
import 'package:flutter/material.dart';
import 'package:pdf/widgets.dart' as pw;
import 'package:pdf/pdf.dart';
import 'package:printing/printing.dart';

class PreviewScreen extends StatelessWidget {
    final pw.Document doc;

    const PreviewScreen({
        Key? key,
        required this.doc,
    }) : super(key: key);

    @override
    Widget build(BuildContext context) {
        return Scaffold(
            appBar: AppBar(
                leading: IconButton(
                    onPressed: () => Navigator.pop(context),
                    icon: const Icon(Icons.arrow_back_outlined),
                ),
                centerTitle: true,
                title: const Text("Preview"),
            ),
            body: PdfPreview(
                build: (format) => doc.save(),
                allowSharing: true,
                allowPrinting: true,
                initialPageFormat: PdfPageFormat.a4,
            ),
        );
    }
}
```

367
368

Gambar 41 Preview Screen



369
370 23. Kemudian buat file **pdf_view.dart** di dalam folder **lib** untuk create pdf
371 nya yang berguna untuk **membuat tampilan dokumen pdfnya**.
372 Penulisan kode dimulai dengan melakukan import hal-hal yang
373 dibutuhkan termasuk librarynya beserta membuat **fungsi createPDF**
374 yang menggunakan **future** yang mana fungsi ini dipanggil pada
375 button di file **home_page.dart**.

```
import 'dart:io';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:pdf/pdf.dart';
import 'package:pdf/widgets.dart' as pw;
import 'package:intl/intl.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/get_total_invoice.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/model/custom_row_invoice.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/model/product.dart';
import 'package:barcode_widget/barcode_widget.dart';
import 'package:gdlibrary_kelas_xxxx/preview_screen.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/item_doc.dart';
import 'package:responsive_sizer/responsive_sizer.dart';

Future<void> createPdf() async {
```

376
377 Gambar 42 Permulaan Mengisi Kode pada Fungsi createPdf dalam File
378 pdf_view.dart

379
380 24. Kemudian, mengisi **parameter** untuk fungsinya sehingga data-data
381 yang dibutuhkan dapat terkirim ke dalam fungsi untuk proses
382 pembuatan dokumen PDF, menjadi sebagai berikut.



The screenshot shows a Flutter application window with a black background and white text. At the top, there are three colored circles (red, yellow, green) representing window controls. Below them is a block of Dart code:

```
import 'dart:io';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:pdf/pdf.dart';
import 'package:pdf/widgets.dart' as pw;
import 'package:intl/intl.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/get_total_invoice.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/model/custom_row_invoice.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/model/product.dart';
import 'package:barcode_widget/barcode_widget.dart';
import 'package:gdlibrary_kelas_xxxx/preview_screen.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/item_doc.dart';
import 'package:responsive_sizer/responsive_sizer.dart';

Future<void> createPdf()
{
    TextEditingController nameController,
    TextEditingController phoneController,
    TextEditingController addressController,
    String id,
    File image,
    BuildContext context,
    List<Product> soldProducts
} async {

}
```

383

384

Gambar 43 Parameter Fungsi createPdf

385

386 25. Setelah itu, memberikan deklarasi hal-hal yang akan digunakan di
387 dalam fungsinya untuk tampilan PDF nya



```
import 'dart:io';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:pdf/pdf.dart';
import 'package:pdf/widgets.dart' as pw;
import 'package:intl/intl.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/get_total_invoice.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/model/custom_row_invoice.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/model/product.dart';
import 'package:barcode_widget/barcode_widget.dart';
import 'package:gdlibrary_kelas_xxxx/preview_screen.dart';
import 'package:gdlibrary_kelas_xxxx/invoice/item_doc.dart';
import 'package:responsive_sizer/responsive_sizer.dart';

Future<void> createPdf
(
    TextEditingController nameController,
    TextEditingController phoneController,
    TextEditingController addressController,
    String id,
    File image,
    BuildContext context,
    List<Product> soldProducts
) async {
    final doc = pw.Document();
    final now = DateTime.now();
    final formattedDate = DateFormat('yyyy-MM-dd HH:mm:ss').format(now); //menampilkan waktu terbuatnya dokumen PDF

    //mengambil gambar dari asset untuk ditampilkan pada dokumen PDF untuk header dokumen pdf dan logo untuk invoice
    final imageLogo =
        (await rootBundle.load("assets/logo.png")).buffer.asUint8List();
    final imageInvoice = pw.MemoryImage(imageLogo);

    //mengambil gambar dari input galeri atau kamera
    pw.ImageProvider pdfImageProvider(Uint8List imageBytes) {
        return pw.MemoryImage(imageBytes);
    }

    final imageBytes = image.readAsBytesSync();

    //Memberi border pada dokumen pdf
    final pdfTheme = pw.PageTheme(
        pageFormat: PdfPageFormat.a4,
        buildBackground: (pw.Context context) {
            return pw.Container(
                decoration: pw.BoxDecoration(
                    border: pw.Border.all(
                        color: PdfColor.fromHex('#FFBD59'),
                        width: 1.w,
                    ),
                ),
            );
        },
    );
};

});
```

388

389

Gambar 44 Deklarasi dalam Fungsi createPdf



```
//peletakan urutan hasil isi atau value baris invoice sesuai yang telah diinputkan menggunakan list
//bentuk widgetnya diatur pada file item_doc.dart
final List<CustomRow> elements = [
    CustomRow("Item Name", "Item Price", "Amount", "Sub Total Product"),
    for (var product in soldProducts)
        CustomRow(
            product.name,
            product.price.toStringAsFixed(2),
            product.amount.toString(),
            (product.price * product.amount).toStringAsFixed(2),
        ),
    CustomRow(
        "Sub Total",
        "",
        "",
        "Rp ${getSubTotal(soldProducts)}", //memanggil fungsi yang ada pada file get_total_invoice.dart
    ),
    CustomRow(
        "PPN Total(11%)",
        "",
        "",
        "Rp ${getPPNTotal(soldProducts)}", //memanggil fungsi yang ada pada file get_total_invoice.dart
    ),
    CustomRow(
        "Total",
        "",
        "",
        "Rp ${((double.parse(getSubTotal(soldProducts)) +
double.parse(getPPNTotal(soldProducts))).toStringAsFixed(2)}", //memanggil fungsi yang ada pada file get_total_invoice.dart
    )
];
pw.Widget table = itemColumn(elements);
}
```

390

391 Gambar 45 Deklarasi untuk Memberi Value pada Tabel Invoice

392

393 26.Tambahkan **doc.addPage()**, seperti ini:

```
,
```

Tambahkan ini, bagian ini yang akan mengatur tampilan halaman dokumen PDF nya

```
"Rp ${((double.parse(getSubTotal(soldProducts)) +
double.parse(getPPNTotal(soldProducts))).toStringAsFixed(2)}", //memanggil fungsi yang ada pada file get_total_invoice.dart
);
];

pw.Widget table = itemColumn(elements);

//Pengaturan tampilan dokumen PDFnya
// Di sini menggunakan Multi Page agar dapat menampilkan lebih dari 1 halaman.
//kemudian menggunakan header untuk header dokumen PDF, build untuk isi dokumen PDFnya,
//dan footer untuk footer dokumen PDF
doc.addPage()
}
```

394

395 Gambar 46 Menambah doc.addPage()



396
397
398

27. Juga tambahkan navigator agar bisa tampil untuk dilihat melalui preview screen seperti yang ada pada tanda kotak merah.

```
//Pengaturan tampilan dokumen PDFnya
//Di sini menggunakan Multi Page agar dapat menampilkan lebih dari 1
halaman.
//kemudian menggunakan header untuk header dokumen PDF, build untuk isi
dokumen PDFnya, dan footer untuk footer dokumen PDF

    doc.addPage(); // Page

    Navigator.push(
        context,
        MaterialPageRoute(
            builder: (context) => PreviewScreen(doc: doc),
        )
    );
}
```

399

400 Gambar 47 Menambah Navigator Setelah doc.addPage agar Tampil pada Preview Screen

401
402

28. Kemudian, **isi kode di dalam doc.addPage**, sebagai berikut.



```
pw.MultiPage(
    pageTheme: pdfTheme,
    header: (pw.Context context) {
        //header PDF nya
        return headerPDF();
    },
    //Isi halaman PDF nya
    build: (pw.Context context) {
        return [
            pw.Center(
                child: pw.Column(
                    mainAxisAlignment: pw.MainAxisAlignment.center,
                    crossAxisAlignment: pw.CrossAxisAlignment.center,
                    children: [
                        pw.Container(
                            margin: pw.EdgeInsets.symmetric(
                                horizontal: 2.h, vertical: 2.h),
                            //Bagian menampilkan gambar yang telah diinput
                            imageFromInput(pdfImageProvider, imageBytes),

                            //Bagian halaman PDF yang menampilkan data personal berupa sebuah
                            tabel yang sudah diinputkan pada halaman input
                            personalDataFromInput(nameController,
                                phoneController, addressController),
                            pw.SizedBox(height: 10.h), //white space untuk jeda

                            // Bagian PDF yang menampilkan invoice
                            topOfInvoice(imageInvoice),
                            barcodeGaris(id),
                            pw.SizedBox(height: 5.h),

                            contentOfInvoice(table),

                            //barcode di dalam dokumen PDF
                            barcodeKotak(id),
                            pw.SizedBox(height: 1.h),
                        )));
        ];
    },
    //Bagian footer
    footer: (pw.Context context) {
        return pw.Container(
            color: PdfColor.fromHex('#FFBD59'),
            child: footerPDF(formattedDate));
    },
);
```

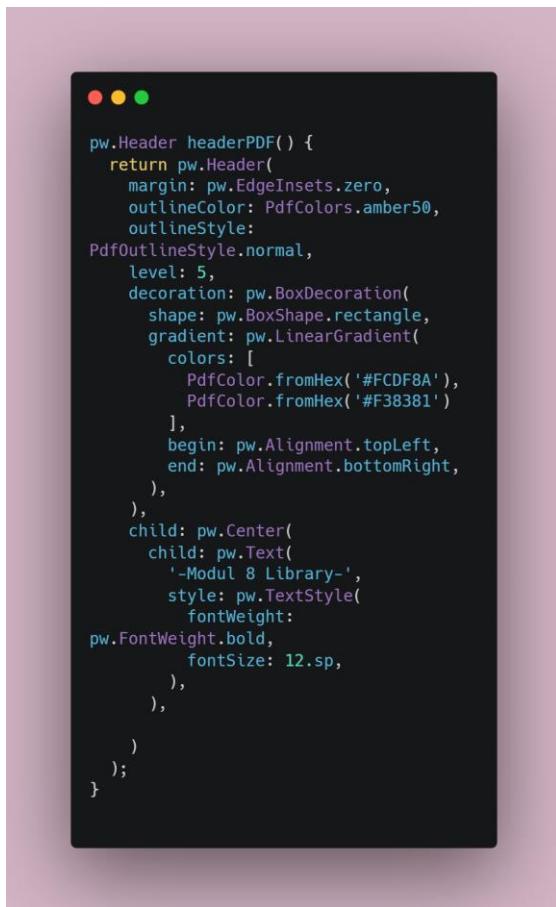
403
404

Gambar 48 Isi Dalam Kurungnya doc.addPage



405 Kemudian, taruh kode kode ini di bawah kurung kurawal tutup dari
406 fungsi createPDF:

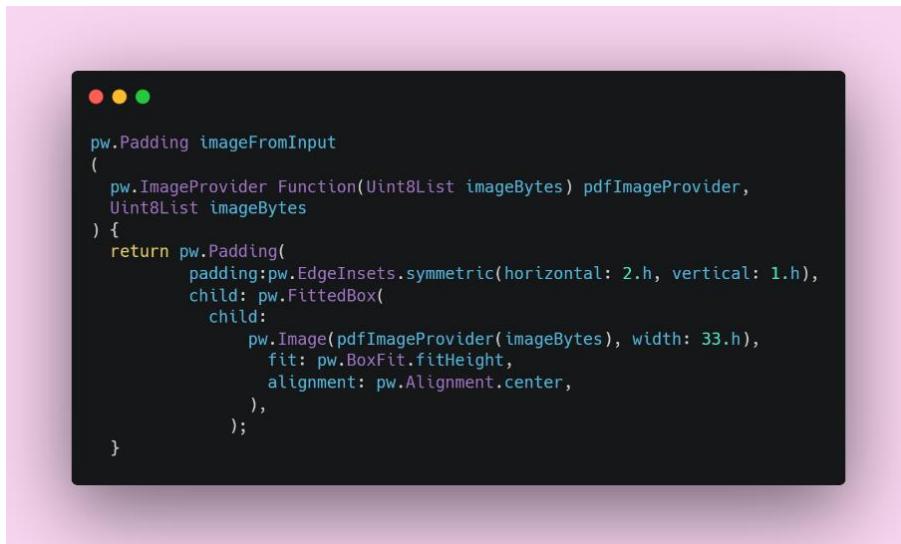
407 **HeaderPDF**



408

409 Gambar 49 Header PDF

410 **imageFromInput**



411

412 Gambar 50 Image From Input



413

414

personalDataFromInput

```
pw.Padding personalDataFromInput(TextEditingController nameController, TextEditingController phoneController,
    TextEditingController addressController) {
  return pw.Padding(
    padding: pw.EdgeInsets.symmetric(horizontal: 5.h, vertical: 1.h),
    child: pw.Table(
      border: pw.TableBorder.all(),
      children: [
        pw.TableRow(
          children: [
            pw.Padding(
              padding: pw.EdgeInsets.symmetric(
                horizontal: 1.h, vertical: 1.h),
              child: pw.Text(
                'Name',
                style: pw.TextStyle(
                  fontWeight: pw.FontWeight.bold,
                  fontSize: 10.sp,
                ),
              ),
            ),
            pw.Padding(
              padding: pw.EdgeInsets.symmetric(
                horizontal: 1.h, vertical: 1.h),
              child: pw.Text(
                nameController.text,
                style: pw.TextStyle(
                  fontWeight: pw.FontWeight.bold,
                  fontSize: 10.sp,
                ),
              ),
            ),
          ],
        ),
        pw.TableRow(
          children: [
            pw.Padding(
              padding: pw.EdgeInsets.symmetric(
                horizontal: 1.h, vertical: 1.h),
              child: pw.Text(
                'Phone Number',
                style: pw.TextStyle(
                  fontWeight: pw.FontWeight.bold,
                  fontSize: 10.sp,
                ),
              ),
            ),
            pw.Padding(
              padding: pw.EdgeInsets.symmetric(
                horizontal: 1.h, vertical: 1.h),
              child: pw.Text(
                phoneController.text,
                style: pw.TextStyle(
                  fontWeight: pw.FontWeight.bold,
                  fontSize: 10.sp,
                ),
              ),
            ),
          ],
        ),
      ],
    ),
  );
}
```

415

416

Gambar 51 Bagian 1 Personal Data Input

417



```
pw.TableRow(  
    children: [  
        pw.Padding(  
            padding: pw.EdgeInsets.symmetric(  
                horizontal: 1.h, vertical: 1.h),  
            child: pw.Text(  
                'Address',  
                style: pw.TextStyle(  
                    fontWeight: pw.FontWeight.bold,  
                    fontSize: 10.sp,  
                ),  
            ),  
        ),  
        pw.Padding(  
            padding: pw.EdgeInsets.symmetric(  
                horizontal: 1.h, vertical: 1.h),  
            child: pw.Text(  
                addressController.text,  
                style: pw.TextStyle(  
                    fontWeight: pw.FontWeight.bold,  
                    fontSize: 10.sp,  
                ),  
            ),  
        ),  
    ],  
    ),  
);  
}
```

418

419

Gambar 52 Bagian 2 Personal Data Input

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437



438

439 **topOfInvoice**

```
pw.Padding topOfInvoice(pw.MemoryImage imageInvoice) {
    return pw.Padding(
        padding: const pw.EdgeInsets.all(8.0),
        child: pw.Row(
            mainAxisAlignment: pw.MainAxisAlignment.spaceBetween,
            children: [
                pw.Image(imageInvoice, height: 30.h, width: 30.w),
                pw.Expanded(child:
                    pw.Container(
                        height: 10.h,
                        decoration: const pw.BoxDecoration(borderRadius:
                            pw.BorderRadius.all(pw.Radius.circular(2)),color: PdfColors.amberAccent),
                        padding: const pw.EdgeInsets.only(left: 40, top: 10, bottom: 10, right: 40),
                        alignment: pw.Alignment.centerLeft,
                        child:
                            pw.DefaultTextStyle(
                                style: const pw.TextStyle(color: PdfColors.amber100, fontSize: 12),
                                child: pw.GridView(
                                    crossAxisCount: 2,
                                    children: [
                                        pw.Text('Awesome Product',
                                            style: pw.TextStyle( fontSize: 10.sp, color: PdfColors.blue800)),
                                        pw.Text('Anggerek Street 12',
                                            style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)),
                                        pw.SizedBox(height: 1.h),
                                        pw.Text('Jakarta 5111',
                                            style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)),
                                        pw.SizedBox(height: 1.h),
                                        pw.SizedBox(height: 1.h),
                                        pw.Text('Contact Us',
                                            style: pw.TextStyle( fontSize: 10.sp, color: PdfColors.blue800)),
                                        pw.SizedBox(height: 1.h),
                                        pw.Text('Phone Number',
                                            style: pw.TextStyle( fontSize: 10.sp, color: PdfColors.blue800)),
                                        pw.Text('0812345678',
                                            style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)),
                                        pw.Text('Email',
                                            style: pw.TextStyle( fontSize: 10.sp, color: PdfColors.blue800)),
                                        pw.Text('awesomeproduct@gmail.com',
                                            style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)),
                                    ],
                                ),
                            ),
                        ],
                    ),
                ),
            ],
        );
}
```

440

441 Gambar 53 Bagian Atas Invoice

442

443

444

445

446

447

448

449



450

contentOfInvoice(XXXX diganti 4 digit NPM Kalian)

```
pw.Padding contentOfInvoice(pw.Widget table) {
  return pw.Padding(
    padding: const pw.EdgeInsets.all(8.0),
    child: pw.Column(children: [
      pw.Text(
        "Dear Customer, thank you for buying our product, we hope the products can make your day."),
      pw.SizedBox(height: 3.h),
      table, //Tabel yang sudah diatur tampilannya di dalam file item_doc.dart
      pw.Text("Thanks for your trust, and till the next time."),
      pw.SizedBox(height: 3.h),
      pw.Text("Kind regards,"),
      pw.SizedBox(height: 3.h),
      pw.Text("XXXX"),
    ]));
}
```

451

452

Gambar 54 Bagian Isi Invoice

453

454

barcodeKotak

```
pw.Padding barcodeKotak(String id) {
  return pw.Padding(
    padding:
      pw.EdgeInsets.symmetric(horizontal: 1.h, vertical: 1.h),
    child: pw.Center(
      child:
        //jenis barcode berbeda dengan value sama(hanya untuk contoh untuk guided)
        pw.BarcodeWidget(
          barcode: pw.Barcode.qrCode(
            errorCorrectLevel: BarcodeQRCorrectionLevel.high,
          ),
          data: id,
          width: 15.w,
          height: 15.h,
        ),
    ),
  );
}
```

455

456

Gambar 55 Barcode Kotak

457

458

459



460

barcodeGaris

```
pw.Container barcodeGaris(String id) {
    return pw.Container(
        child:
            pw.Padding(
                padding: pw.EdgeInsets.symmetric(
                    horizontal: 1.h, vertical: 1.h),
                child: pw.BarcodeWidget(
                    barcode: Barcode.code128(escapes:
true),      data: id,
                    width: 10.w,
                    height: 5.h,
                ),
            ),
    );
}
```

461

Gambar 56 Barcode Garis

463

464

footerPDF

```
pw.Center footerPDF(String formattedDate) =>
    pw.Center(child: pw.Text('Created At $formattedDate', style: pw.TextStyle( fontSize: 10.sp, color: PdfColors.blue)));
```

465

Gambar 57 Footer PDF

467

468

469

470

471

472

473



474 Maka, setelah selesai membuat guided jika di run akan tampil seperti ini:

Modul 8 Library

Fill This Form

Pick Image from Camera

Pick Image from Gallery

No image selected

Name

Phone Number

Address

Buy Product

Item	Price	+	0	-
Membership	Price: Rp 50000.00	+ 0	0	-
Books	Price: Rp 30000.00	+ 0	0	-
Pen	Price: Rp 5000.00	+ 0	0	-
Snacks	Price: Rp 10000.00	+ 0	0	-

475

476 Gambar 58 Hasil Jadi Bagian Atas Home Page

477

Modul 8 Library

Pick Image from Gallery

No image selected

Name

Phone Number

Address

Buy Product

Item	Price	+	0	-
Membership	Price: Rp 50000.00	+ 0	0	-
Books	Price: Rp 30000.00	+ 0	0	-
Pen	Price: Rp 5000.00	+ 0	0	-
Snacks	Price: Rp 10000.00	+ 0	0	-

PPN(11%)
Total

Rp 0.00
Rp 0.00 €

Create PDF

478

479 Gambar 59 Hasil Jadi Bagian Bawah Home Page



480

481 Ini hasilnya jika kalian input gambar

Modul 8 Library

Fill This Form

[Pick Image from Camera](#)

[Pick Image from Gallery](#)


✖

Name

Birthday

Age

482

483 Gambar 60 Hasil Jadi Jika Input Gambar

484 Ini hasilnya jika kalian input textfield

Modul 8 Library

Fill This Form

[Pick Image from Camera](#)

[Pick Image from Gallery](#)


✖

Name
Jessica

Phone Number
08123456789

Address
A Street 12

485

486 Gambar 61 Hasil Jadi Jika Input Gambar dan Text Field



487

Modul 8 Library

Pick Image from Gallery

No image selected

Name _____

Phone Number _____

Address _____

Membership Price: Rp 50000.00 + 0 -

Books Price: Rp 30000.00 + 0 -

Pen Price: Rp 5000.00 + 0 -

Snacks Price: Rp 10000.00 + 0 -

Hamburger Price: Rp 25000.00 + 0 -

PPN(11%) Total Rp 0.00 Rp 0.00

Warning

Please fill in all the fields.

OK

Create PDF

488

489 Gambar 62 Hasil Jadi Warning Jika Ada Text Field yang Belum Diisi

490

491

Modul 8 Library




Name
Jessica

Phone Number
08123456789

Address
A Street 12

Buy Product

Product	Price: Rp 50000.00	+	-	Total
Membership	Price: Rp 50000.00	+	-	Rp 13200.00
Books	Price: Rp 30000.00	+	-	Rp 12000.00
Pen	Price: Rp 5000.00	+	-	
Snacks	Price: Rp 10000.00	+	-	
Hamburger	Price: Rp 25000.00	+	-	
PPN(11%)				
Total				Rp 12000.00

Create PDF

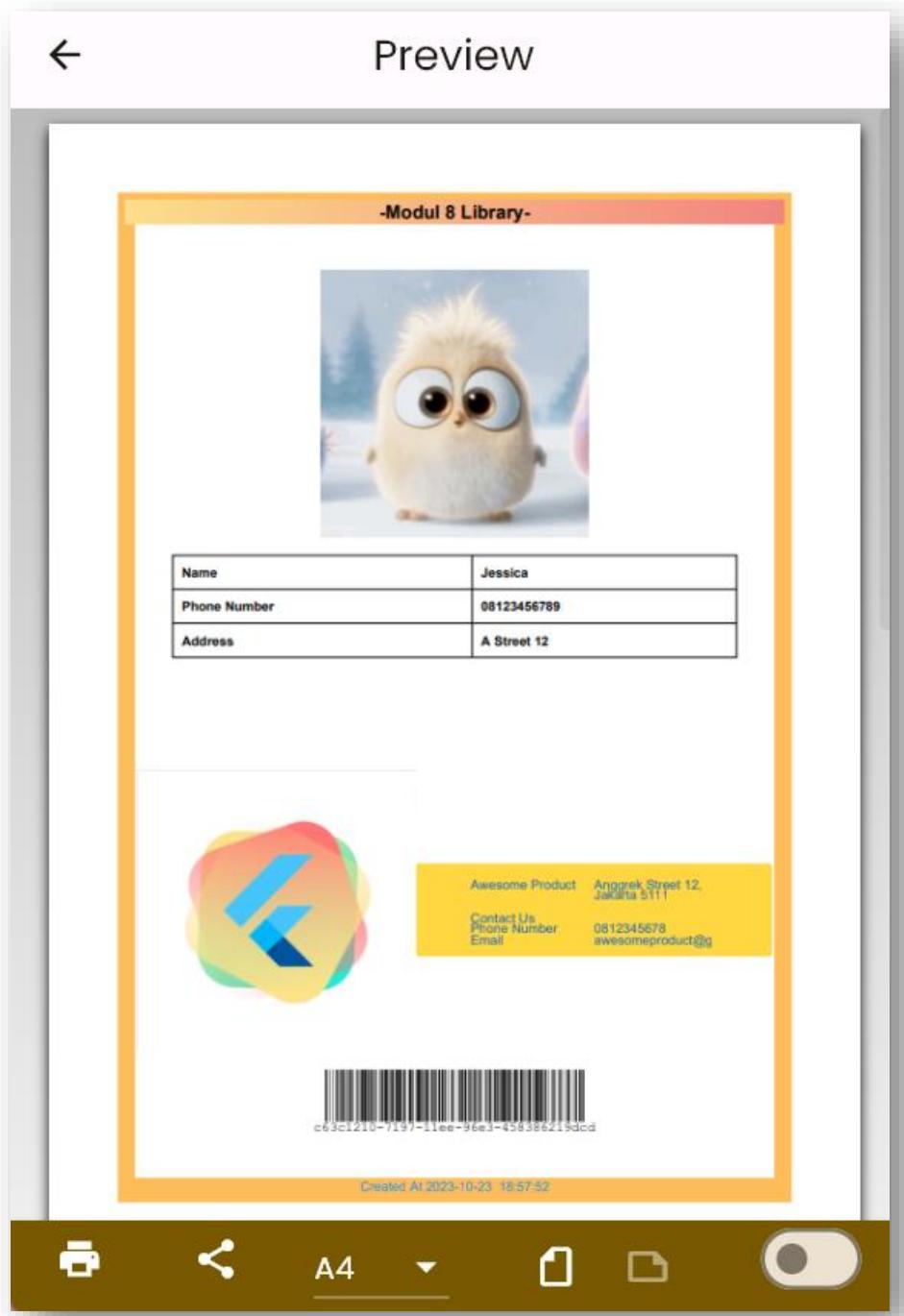
492

493 Gambar 63 Hasil Jadi Ketika Input Semuanya pada Home Page



494

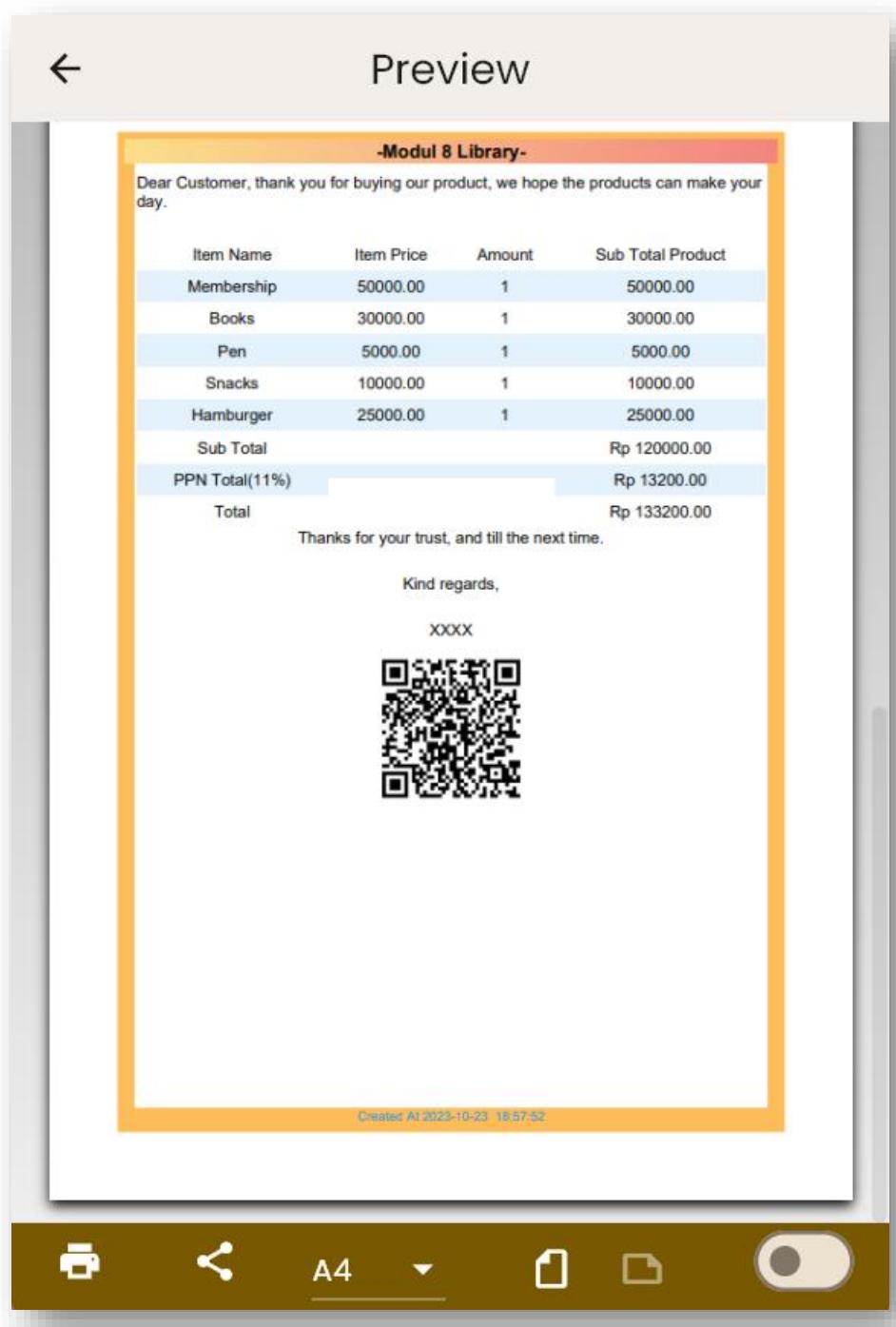
Kemudian setelah klik button create PDF nya menjadi seperti ini:



495

496

Gambar 64 Hasil Jadi Preview dengan Tampilan Dokumen PDF Bagian Atas



497

498 Gambar 65 Hasil Jadi Preview dengan Tampilan Dokumen PDF Bagian Bawah

499

500

501

502

503
504

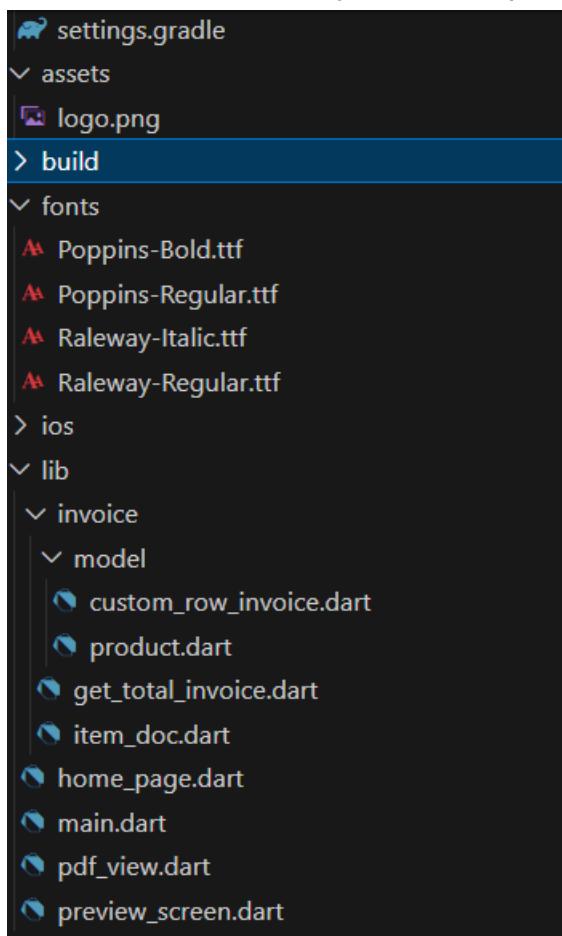
PEMBAHASAN GUIDED MODUL 8

505
506
507

Pada guided 1 di atas, anda telah mencoba untuk membuat sebuah aplikasi pembuat dokumen PDF berdasarkan input dengan menggunakan Visual Studio Code, sebagai berikut pembahasannya.

508
509
510
511
512
513
514

1. Jika menggunakan sizer, maka gunakan .sp, .h, .w, dll sesuai dokumentasi yang ada pada link
2. Perhatikan error message, resiko error ada pada salah peletakan const, tanda tutup kurung yang kurang atau berlebih
3. Selengkapnya mengenai custom font pada link:
<https://docs.flutter.dev/cookbook/design/fonts>
4. Jika dilihat total filenya akan seperti ini:



515

516
517
518

Gambar 66 Total Folder dan File yang Terbuat

5. Terdapat 3 macam bentuk tabel dengan cara pembuatan yang berbeda, antara lain:



519

1. Pada **personalDataFromInput** pada file **pdf_view.dart**

520

521 Gambar 67 Hasil Tampilan Tabel Data Personal pada Dokumen PDF

522

2. Pada **topOfInvoice** pada file **pdf_view.dart**

523

```
pw.Expanded(child:  
  pw.Container(  
    height: 10.h,  
    decoration: const pw.BoxDecoration(borderRadius: pw.BorderRadius.all(pw.Radius.circular(2)),color: PdfColors.amberAccent),  
    padding: const pw.EdgeInsets.only(left: 40, top: 10, bottom: 10, right: 40),  
    alignment: pw.Alignment.centerLeft,  
    child:  
      pw.DefaultTextStyle(  
        style: const pw.TextStyle(color: PdfColors.amber100, fontSize: 12),  
        child: pw.GridView(  
          crossAxisCount: 2,  
          children: [  
            pw.Text('Awesome Product',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
            pw.Text('Anggrek Street 12',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
            pw.SizedBox(height: 1.h),  
            pw.Text('Jakarta 5111',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
            pw.SizedBox(height: 1.h),  
            pw.SizedBox(height: 1.h),  
            pw.Text('Contact Us',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
            pw.SizedBox(height: 1.h),  
            pw.Text('Phone Number',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
            pw.Text('0812345678',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
            pw.Text('Email',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
            pw.Text('awesomeproduct@gmail.com',  
              style: pw.TextStyle(fontSize: 10.sp, color: PdfColors.blue800)), // pw.Text  
          ],  
        ), // pw.GridView  
      ), // pw.DefaultTextStyle  
    ), // pw.Container  
  ), // pw.Expanded
```

524

525 Gambar 68 Kode di Dalam **topOfInvoice** untuk Membuat Informasi
526 Berupa Tabel Tanpa Border

Awesome Product	Anggrek Street 12
Jakarta 5111	
Contact Us	
Phone Number	0812345678
Email	awesomeproduct@g

527

528 Gambar 69 Hasil Jadi Bagian Atas Invoice yang Berupa Tabel

529 3. Pada **contentOfInvoice** pada file **pdf_view.dart** dengan **value**
 530 yang diatur pada **elements** beserta **tampilan** tabelnya yang
 531 diatur pada file **item_doc.dart**. File **item_doc.dart** nya bisa dilihat
 532 di atas atau pada file yang sudah kalian buat masing-masing.
 533 Kemudian terdapat deklarasi pw. Widget

Item Name	Item Price	Amount	Sub Total Product
Membership	50000.00	1	50000.00
Books	30000.00	1	30000.00
Pen	5000.00	1	5000.00
Snacks	10000.00	1	10000.00
Hamburger	25000.00	1	25000.00
Sub Total		Rp 120000.00	
PPN Total(11%)		Rp 13200.00	
Total		Rp 133200.00	

534

Gambar 70 Hasil Jadi Tabel Invoice dengan Warna Row

536

```
//pemotongan urutan hasil isi atau value baris invoiceanya sesuai yang telah diinputkan menggunakan list
//bentuk widgetnya diatur pada file item_doc.dart
final List<CustomRow> elements = [
  CustomRow("Item Name", "Item Price", "Amount", "Sub Total Product"),
  for (var product in soldProducts)
    CustomRow(
      product.name,
      product.price.toStringAsFixed(2),
      product.amount.toString(),
      (product.price * product.amount).toStringAsFixed(2),
    ),
  CustomRow(
    "Sub Total",
    "",
    "",
    "Rp ${getSubTotal(soldProducts)}", //memanggil fungsi yang ada pada file get_total_invoice.dart
  ),
  CustomRow(
    "PPN Total(11%)",
    "",
    "",
    "Rp ${getPPNTotal(soldProducts)}", //memanggil fungsi yang ada pada file get_total_invoice.dart
  ),
  CustomRow(
    "Total",
    "",
    "",
    "Rp ${((double.parse(getSubTotal(soldProducts)) + double.parse(getPPNTotal(soldProducts))).toStringAsFixed(2))}", //memanggil fungsi yang ada pada file get_total_invoice.dart
  )
];
pw.Widget table = itemColumn(elements);
```

537

Gambar 71 Deklarasi Value dan Tampilan dari File item_doc.dart di File pdf_view.dart dengan Value dari elements

540

```
pw.Padding contentOfInvoice(pw.Widget table) {
  return pw.Padding(
    padding: const pw.EdgeInsets.all(8.0),
    child: pw.Column(children: [
      pw.Text(
        "Dear Customer, thank you for buying our product, we hope the products can make your day."),
      pw.SizedBox(height: 3.h),
      table, //Tabel yang sudah diatur tampilannya di dalam file item_doc.dart
      pw.Text("Thanks for your trust, and till the next time."),
      pw.SizedBox(height: 3.h),
      pw.Text("Kind regards,"),
      pw.SizedBox(height: 3.h),
      pw.Text("XXXX"),
    ]));
}
```

541

Gambar 72 Pemanggilan table pada contentOfInvoice



543 6. Perbedaan sintaks antara menggunakan library responsive sizer
544 dengan yang tidak:

545

Tanpa Sizer	Sizer
<pre>class MyApp extends StatelessWidget { const MyApp({super.key}); // This widget is the root of your application. @override Widget build(BuildContext context) { return MaterialApp(title: 'Flutter Demo', theme: ThemeData(fontFamily: 'Poppins', colorScheme: ColorScheme.fromSeed(seedColor: Colors.amber) .copyWith(secondary: Colors.amberAccent,), materialTapTargetSize: MaterialTapTargetSize.shrinkWrap, appBarTheme: const AppBarTheme(systemOverlayStyle: SystemUiOverlayStyle.dark,), // AppBarTheme useMaterial3: true,), // ThemeData home: const MyHomePage(),); // MaterialApp } }</pre>	<pre>class MyApp extends StatelessWidget { const MyApp({super.key}); // This widget is the root of your application. @override Widget build(BuildContext context) { return ResponsiveSizer(builder: (context, orientation, deviceType) { if (Device.orientation == Orientation.portrait) return Container(// Widget for Portrait width: 100.w, height: 20.5.h,); else // Container return Container(// Widget for Landscape width: 100.w, height: 12.5.h,); }); Device.screenType == ScreenType.tablet ? Container(// Widget for Tablet width: 100.w, height: 20.5.h,) : Container(// Container width: 100.w, height: 12.5.h,); return MaterialApp(title: 'Flutter Demo', debugShowCheckedModeBanner: false, theme: ThemeData(fontFamily: 'Poppins', colorScheme: ColorScheme.fromSeed(seedColor: Colors.amber) .copyWith(secondary: Colors.amberAccent,), materialTapTargetSize: MaterialTapTargetSize.shrinkWrap, appBarTheme: const AppBarTheme(systemOverlayStyle: SystemUiOverlayStyle.dark,), // AppBarTheme useMaterial3: true,), // ThemeData home: const MyHomePage(),); // MaterialApp } }</pre>
height: 20	height: 20.h
width: 10	width: 10.w
fontSize: 8	fontSize: 8.sp

546

547

548

549

550

551

552

553



554

555

556 **ATURAN PENGERJAAN GUIDED :**

557 - **Guided dikerjakan sebelum waktu perkuliahan berlangsung yang
diberi waktu satu minggu karena modulnya panjang.**

559 - **Penamaan projek guided harus sesuai dengan yang sudah
dicontohkan.**

561 - **Guided dikumpulkan melalui github dengan penamaan setiap file
pada github adalah : *GDLIBRARY_KELAS_XXXX* (contoh :
GDLibrary_A_1234)**

564 **○ NAMAGUIDED → SESUAI DENGAN CONTOH DALAM MODUL INI**

565 **○ XXXX → 4 DIGIT TERAKHIR NPM**

566 - **Setelah diupload melalui github, jangan lupa untuk
mengumpulkan keseluruhan link file github melalui situs kuliah.**

568

569

570

571

572

573

574

575

576 *- Don't stop when you are tired. Stop when you are done -*

577