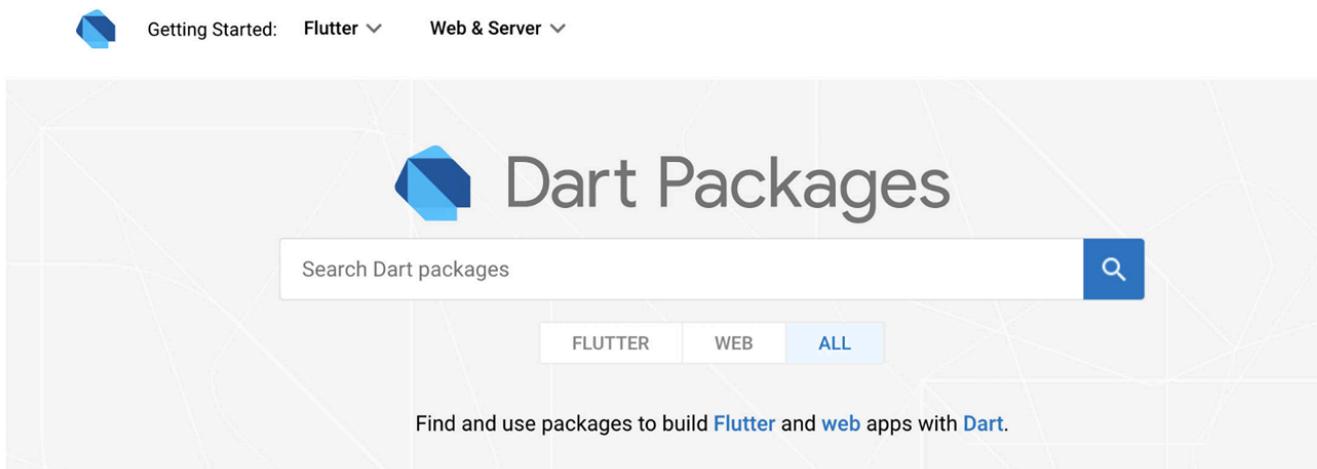


Flutter – Basic Mobile Packages



Top Dart packages

[http](#)
FLUTTER WEB OTHER

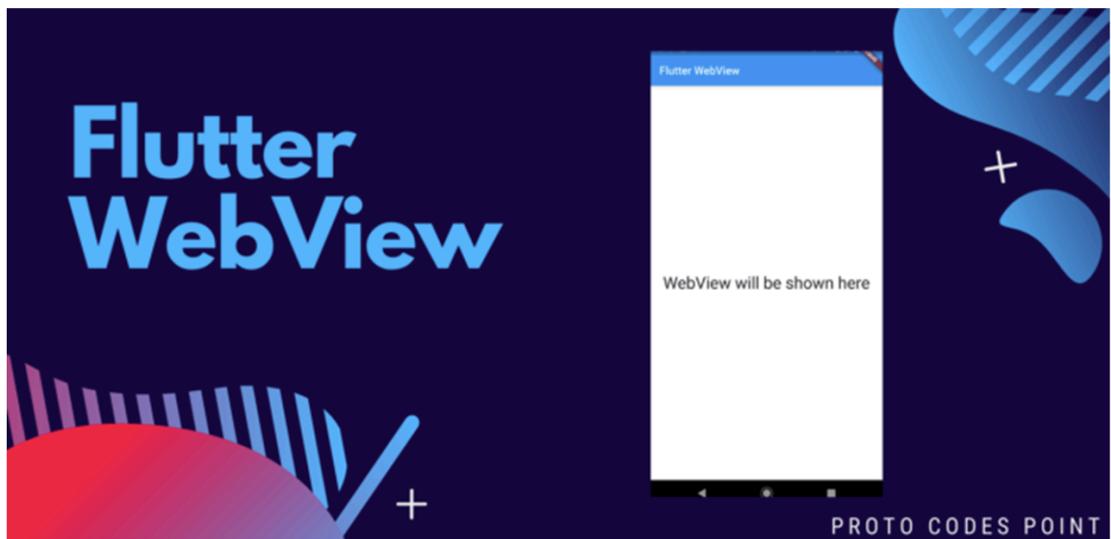
[shared_preferences](#)
FLUTTER

[url_launcher](#)
FLUTTER

Point & Use

Flutter สนับสนุนการใช้งาน Package ที่มีผู้พัฒนารายอื่น ๆ เข้ามาใช้งานร่วมได้ ทำให้สามารถพัฒนาแอปพลิเคชันได้หลากหลายมากขึ้น เช่น การเข้าถึงอินเทอร์เน็ต กล้องถ่ายภาพและวิดีโอ ภาพเคลื่อนไหว พิกัดตำแหน่ง และแผนที่ โดยนำ package มาเพื่อใช้ความสามารถของอุปกรณ์เคลื่อนที่ให้ช่วยอำนวยความสะดวก สะดวกในการพัฒนาแอปพลิเคชันได้

Flutter – WebView



WebView เป็นเครื่องมือที่ทำหน้าที่ดึงข้อมูลจากเว็บไซต์มาแสดงผล

ขั้นตอนการใช้งาน WebView

1. ติดตั้ง package ใน pubspec.yaml

webview_flutter 4.10.0

Published 4 months ago • [flutter.dev](#) Dart 3 compatible

SDK

FLUTTER

PLATFORM

ANDROID

IOS

MACOS

4.2K

Readme

Changelog

Example

Installing

Versions

Scores

Use this package as a library

Depend on it

Run this command:

With Flutter:

```
$ flutter pub add webview_flutter
```

This will add a line like this to your package's pubspec.yaml (and run an implicit flutter pub get):

```
dependencies:  
  webview_flutter: ^4.10.0
```

2. Import WebView

Import it

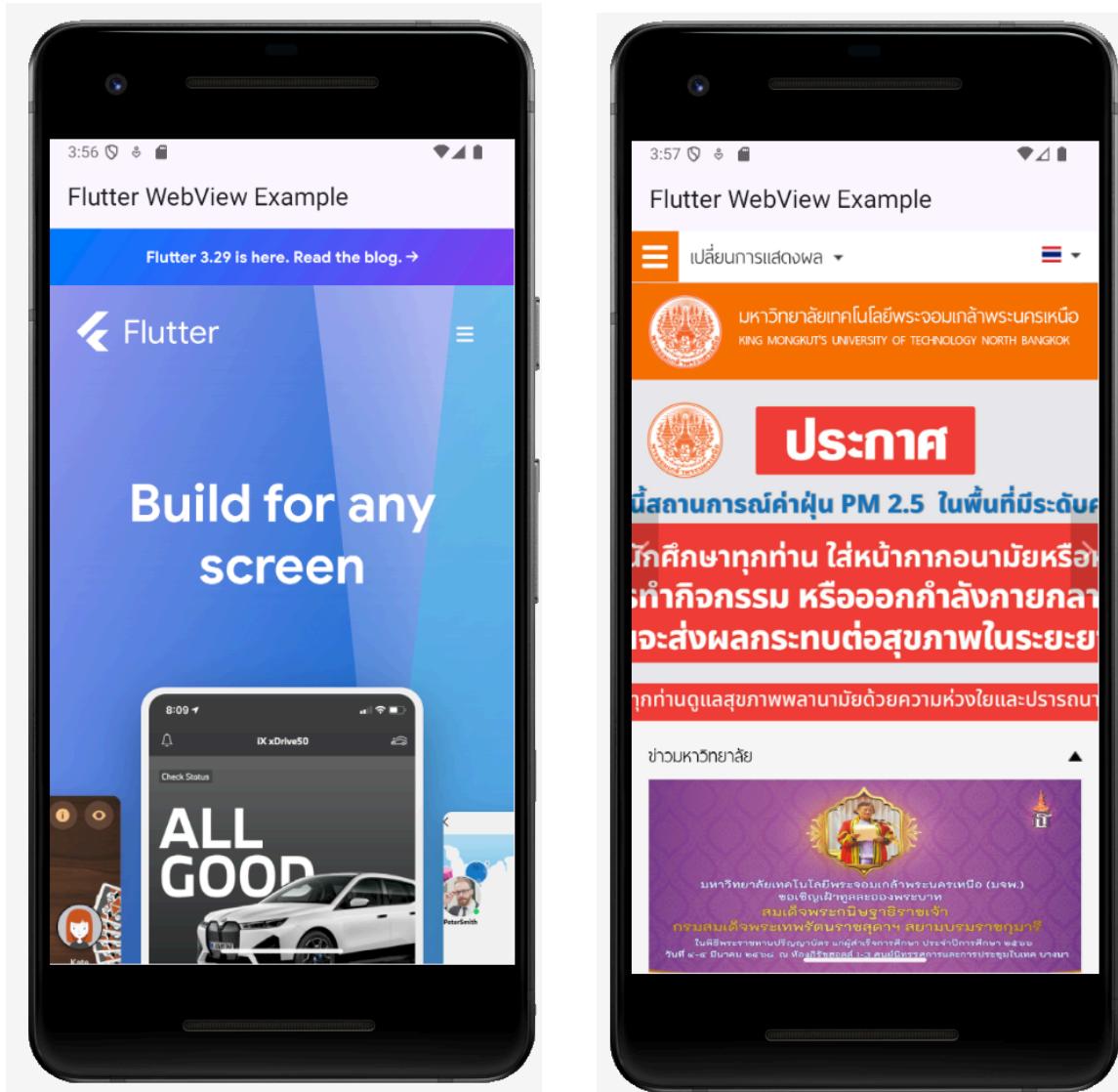
Now in your Dart code, you can use:

```
import 'package:webview_flutter/webview_flutter.dart';
```

3. เรียกใช้งาน WebView โดยกำหนดคุณสมบัติดังนี้

* **initialUrl** คือ การกำหนด url ของเว็บไซต์ที่ต้องการแสดงผลเมื่อเปิด WebView ขึ้นมาครั้งแรก
กำหนดให้อยู่ในรูปแบบ String เช่น “<https://www.google.com>” , <https://flutter.dev>

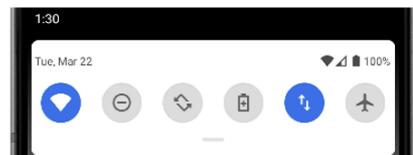
* **javaScriptMode** คือ การกำหนดให้ javaScript สามารถแสดงผลได้บน WebView



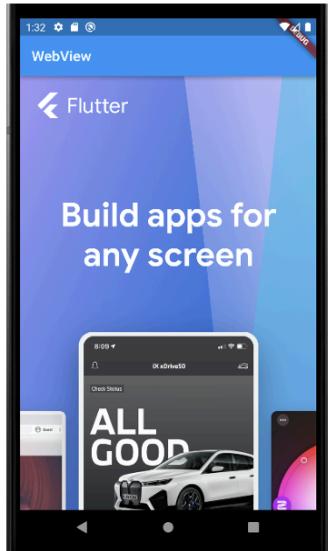


```
1 import 'package:flutter/material.dart';
2 import 'package:webview_flutter/webview_flutter.dart';
3
4 void main() => runApp(
5   const MaterialApp(home: WebViewExample(), debugShowCheckedModeBanner: false),
6 ),
7
8 class WebViewExample extends StatefulWidget {
9   const WebViewExample({super.key});
10
11   @override
12   State<WebViewExample> createState() => _WebViewExampleState();
13 }
14
15 class _WebViewExampleState extends State<WebViewExample> {
16   late final WebViewController _controller;
17
18   @override
19   void initState() {
20     super.initState();
21
22     // Initialize WebView
23     _controller =
24       WebViewController()
25         ..setJavaScriptMode(
26             JavaScriptMode.unrestricted,
27           ) // เปิดใช้งาน JavaScript
28         ..loadRequest(Uri.parse('https://flutter.dev')); // โหลด URL
29   }
30
31   @override
32   Widget build(BuildContext context) {
33     return Scaffold(
34       appBar: AppBar(title: const Text('Flutter WebView Example')),
35       body: WebViewWidget(controller: _controller),
36     );
37   }
38 }
39
```

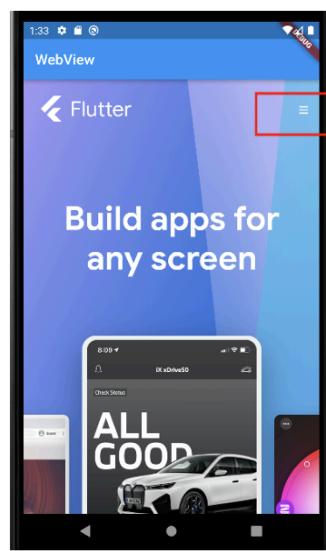
4. เปิดการเชื่อมต่อ Internet ให้กับ Emulator ด้วย



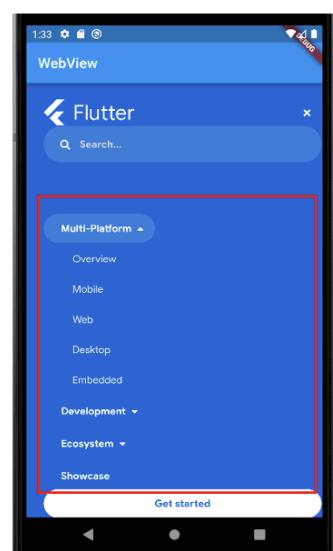
ทดสอบการใช้งาน



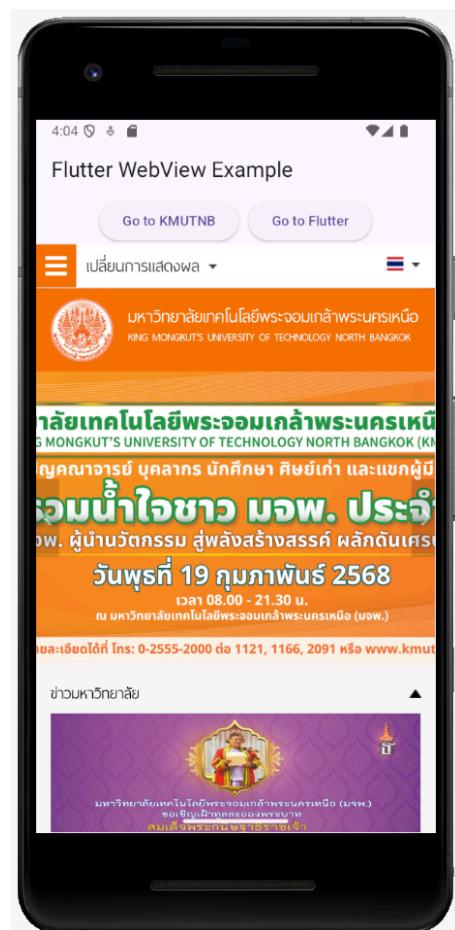
* ไม่ได้กำหนด JavaScriptMode



** กำหนด JavaScriptMode



***** ปรับปรุงการทำงานของ Application
ให้นักศึกษา สร้างปุ่ม เพื่อเลือกการแสดงผล Website ดังภาพด้านล่าง**



```
1 import 'package:flutter/material.dart';
2 import 'package:webview_flutter/webview_flutter.dart';
3
4 void main() => runApp(
5   const MaterialApp(home: WebViewExample(), debugShowCheckedModeBanner: false),
6 );
7
8 class WebViewExample extends StatefulWidget {
9   const WebViewExample({super.key});
10
11   @override
12   State<WebViewExample> createState() => _WebViewExampleState();
13 }
14
15 class _WebViewExampleState extends State<WebViewExample> {
16   late final WebViewController _controller;
17
18   @override
19   void initState() {
20     super.initState();
21
22     // Initialize WebView
23     _controller =
24       WebViewController()
25         ..setJavaScriptMode(
26           JavaScriptMode.unrestricted,
27         ) // เปิดไฟฟ้า JavaScript
28         ..loadRequest(Uri.parse('https://kmutnb.ac.th')); // โหลด URL แรก
29   }
30
31   // ฟังก์ชันที่เปลี่ยน URL ตามปุ่ม
32   void _loadUrl(String url) {
33     _controller.loadRequest(Uri.parse(url)); // โหลด URL ใหม่
34   }
35
36   @override
37   Widget build(BuildContext context) {
38     return Scaffold(
39       appBar: AppBar(title: const Text('Flutter WebView Example')),
40       body: Column(
41         children: [
42           // ปุ่มสำหรับเปลี่ยน URL
43           Row(
44             mainAxisAlignment: MainAxisAlignment.center,
45             children: [
46               ElevatedButton(
47                 onPressed: () {
48                   _loadUrl('https://kmutnb.ac.th'); // กดปุ่มแรก
49                 },
50                 child: const Text('Go to KMUTNB'),
51               ),
52               const SizedBox(width: 10), // ช่องว่างระหว่างปุ่ม
53               ElevatedButton(
54                 onPressed: () {
55                   _loadUrl('https://flutter.dev'); // กดปุ่มที่สอง
56                 },
57                 child: const Text('Go to Flutter'),
58               ),
59             ],
60           ),
61           // WebView แสดงผล
62           Expanded(child: WebViewWidget(controller: _controller)),
63         ],
64       ),
65     );
66   }
67 }
68 }
```

Flutter – QR Code Scanner

QR Code Scanner App



QR Code คือ บาร์โค้ดสองมิติ สามารถอ่านข้อมูลได้แนวตั้งและแนวนอน

ช่วยให้สามารถจัดเก็บข้อมูลได้มากกว่าการໂຄດແບບเดิม

เหมาะสมสำหรับงานที่ต้องการเก็บข้อมูลที่ซับซ้อนมากขึ้นได้

Flutter มีไลบรารี ที่ถูกพัฒนามาให้อ่านรหัส QR Code มากราย

โดยในใบงานนี้ จะแนะนำการใช้งานไลบรารี คือ flutter_qr_bar_scanner

flutter_barcode_scanner_plus 3.0.8

Published 4 days ago Dart 3 compatible

SDK FLUTTER PLATFORM ANDROID IOS

11

Readme Changelog Example Installing Versions Scores

Use this package as a library

Depend on it

Run this command:

With Flutter:

```
$ flutter pub add flutter_barcode_scanner_plus
```

This will add a line like this to your package's pubspec.yaml (and run an implicit flutter pub get):

```
dependencies:  
  flutter_barcode_scanner_plus: ^3.0.8
```

Alternatively, your editor might support flutter pub get. Check the docs for your editor to learn more.

Import it

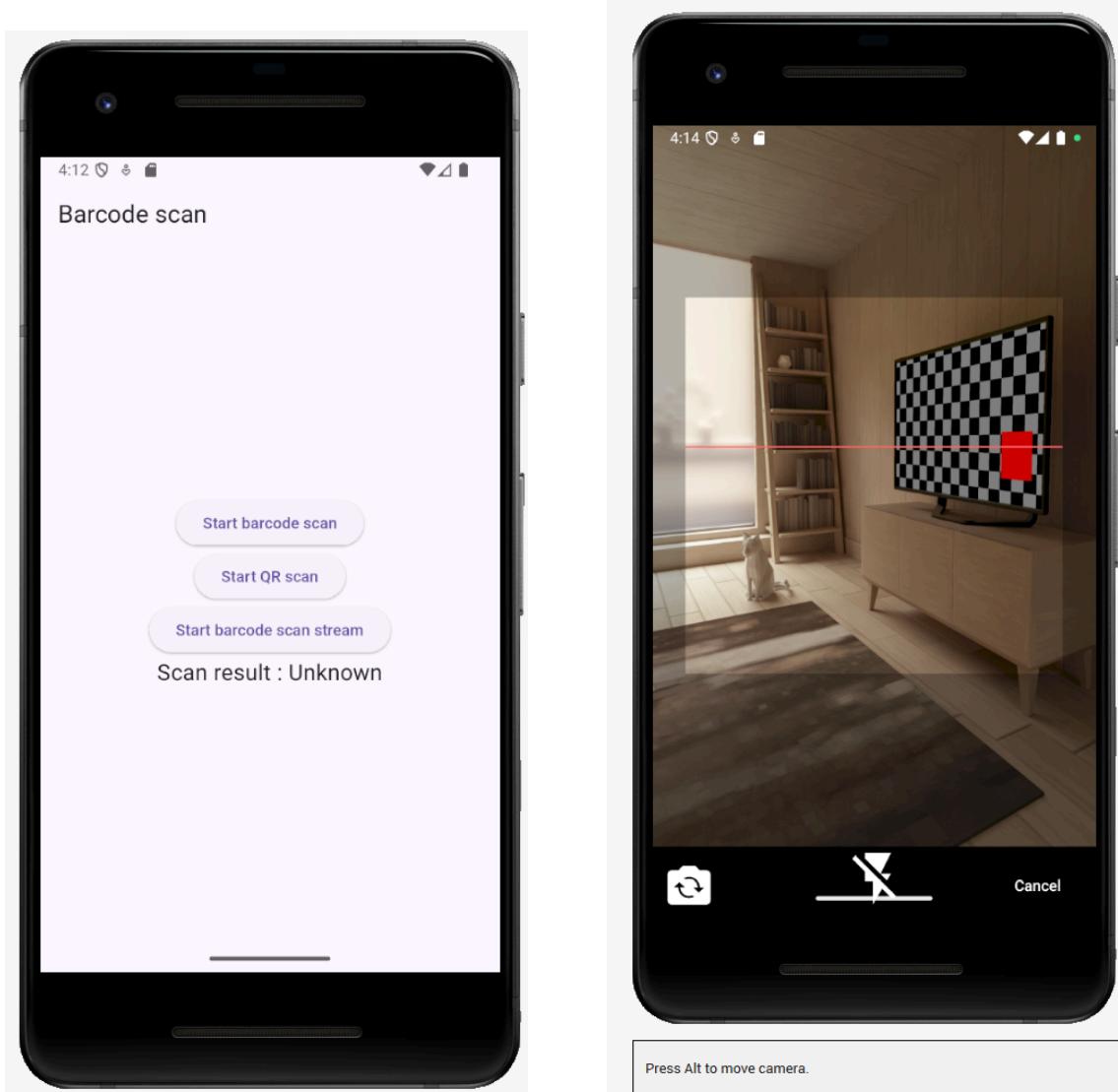
Now in your Dart code, you can use:

```
import 'package:flutter_barcode_scanner_plus/flutter_barcode_scanner_plus.dart';
```

```
lib > main.dart > _MyAppState
1 import 'dart:async';
2
3 import 'package:flutter/material.dart';
4 import 'package:flutter/services.dart';
5 import 'package:flutter_barcode_scanner_plus/flutter_barcode_scanner_plus.dart';
6
7 void main() => runApp(app: MyApp());
8
9 class MyApp extends StatefulWidget {
10   @override
11   _MyAppState createState() => _MyAppState();
12 }
13
14 class _MyAppState extends State<MyApp> {
15   String _scanBarcode = 'Unknown';
16
17   @override
18   void initState() {
19     super.initState();
20   }
21
22   Future<void> startBarcodeScanStream() async {
23     FlutterBarcodeScanner.getBarcodeStreamReceiver(
24       lineColor: '#ff6666',
25       cancelButtonTitle: 'Cancel',
26       isShowFlashIcon: true,
27       scanMode: ScanMode.BARCODE,
28     )!.listen(onData: (dynamic barcode) => print(object: barcode));
29   }
30
31   Future<void> scanQR() async {
32     String barcodeScanRes;
33     // Platform messages may fail, so we use a try/catch PlatformException.
34     try {
35       barcodeScanRes = await FlutterBarcodeScanner.scanBarcode(
36         lineColor: '#ff6666',
37         cancelButtonTitle: 'Cancel',
38         isShowFlashIcon: true,
39         scanMode: ScanMode.QR,
40       );
41       print(object: barcodeScanRes);
42     } on PlatformException {
43       barcodeScanRes = 'Failed to get platform version.';
44     }
45   }
46 }
```

```
45     // If the widget was removed from the tree while the asynchronous platform
46     // message was in flight, we want to discard the reply rather than calling
47     // setState to update our non-existent appearance.
48     if (!mounted) return;
49
50     setState(fn: () {
51         _scanBarcode = barcodeScanRes;
52     });
53 }
54 // Platform messages are asynchronous, so we initialize in an async method.
55 Future<void> scanBarcodeNormal() async {
56     String barcodeScanRes;
57     // Platform messages may fail, so we use a try/catch PlatformException.
58     try {
59         barcodeScanRes = await FlutterBarcodeScanner.scanBarcode(
60             lineColor: '#ff6666',
61             cancelButtonTitle: 'Cancel',
62             isShowFlashIcon: true,
63             scanMode: ScanMode.BARCODE,
64         );
65         print(object: barcodeScanRes);
66     } on PlatformException {
67         barcodeScanRes = 'Failed to get platform version.';
68     }
69     // If the widget was removed from the tree while the asynchronous platform
70     // message was in flight, we want to discard the reply rather than calling
71     // setState to update our non-existent appearance.
72     if (!mounted) return;
73     setState(fn: () {
74         _scanBarcode = barcodeScanRes;
75     });
76 }
77 }
```

```
78 @override
79 Widget build(BuildContext context) {
80     return MaterialApp(
81         debugShowCheckedModeBanner: false,
82         home: Scaffold(
83             appBar: AppBar(title: const Text(data: 'Barcode scan')),
84             body: Builder(
85                 builder: (BuildContext context) {
86                     return Container(
87                         alignment: Alignment.center,
88                         child: Flex(
89                             direction: Axis.vertical,
90                             mainAxisAlignment: MainAxisAlignment.center,
91                             children: <Widget>[
92                                 ElevatedButton(
93                                     onPressed: () => scanBarcodeNormal(),
94                                     child: Text(data: 'Start barcode scan'),
95                                 ), // ElevatedButton
96                                 ElevatedButton(
97                                     onPressed: () => scanQR(),
98                                     child: Text(data: 'Start QR scan'),
99                                 ), // ElevatedButton
100                                ElevatedButton(
101                                    onPressed: () => startBarcodeScanStream(),
102                                    child: Text(data: 'Start barcode scan stream'),
103                                ), // ElevatedButton
104                                Text(
105                                    data: 'Scan result : ${_scanBarcode}\n',
106                                    style: TextStyle(fontSize: 20),
107                                ), // Text
108                                ], // <Widget>[]
109                            ), // Flex
110                        ); // Container
111                    },
112                ), // Builder
113            ), // Scaffold
114        ); // MaterialApp
115    }
116 }
117 }
```



** ให้ทำการ Build Code เพื่อสร้างไฟล์ apk สำหรับติดตั้งบนอุปกรณ์จริง

วิธีการสร้างไฟล์ Android Package (.apk)

***** หากต้องการลบข้อมูลการสร้างแอปพลิเคชัน รวมทั้งลบข้อมูลในหน่วยความจำที่ติดมากจากการใช้งานหรือการทดสอบในขั้นตอนการสร้างแอปพลิเคชัน ให้ใช้คำสั่ง flutter clean

```
PS D:\2564-2\060233209 Mobile App Dev\MobileApp\flutter_qrcode_1> flutter clean
```

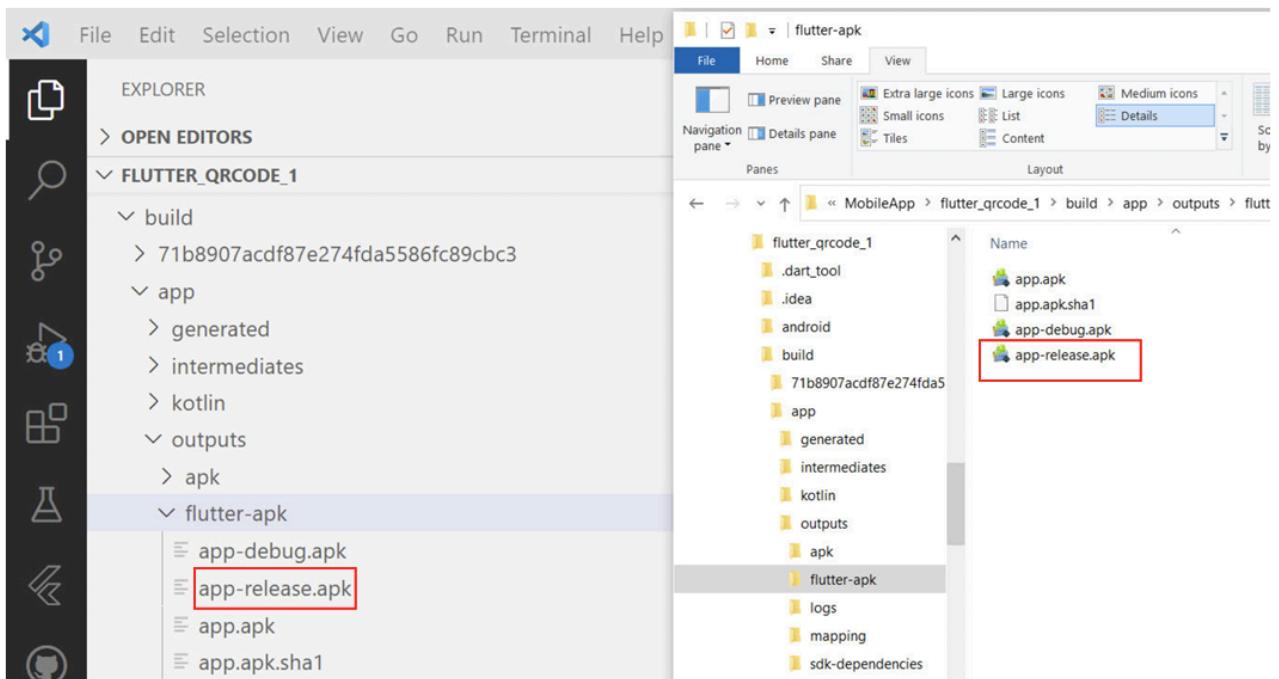
คำสั่งสำหรับ Build apk

```
PS D:\2564-2\060233209 Mobile App Dev\MobileApp\flutter_qrcode_1> flutter build apk
```

ไฟล์ apk ที่ build เสร็จแล้ว

จะอยู่ที่ build\app\outputs\flutter-apk\app-release.apk

```
Running Gradle task 'assembleRelease'...                                153.0s
✓ Built build\app\outputs\flutter-apk\app-release.apk (21.0MB).
PS D:\2564-2\060233209 Mobile App Dev\MobileApp\flutter_qrcode_1>
⑧ 3 △ 0 ① 14 ⌘> Debug my code + packages      UTF-8   CRLF   Dart   Dart DevTools   ⌂ Go
```

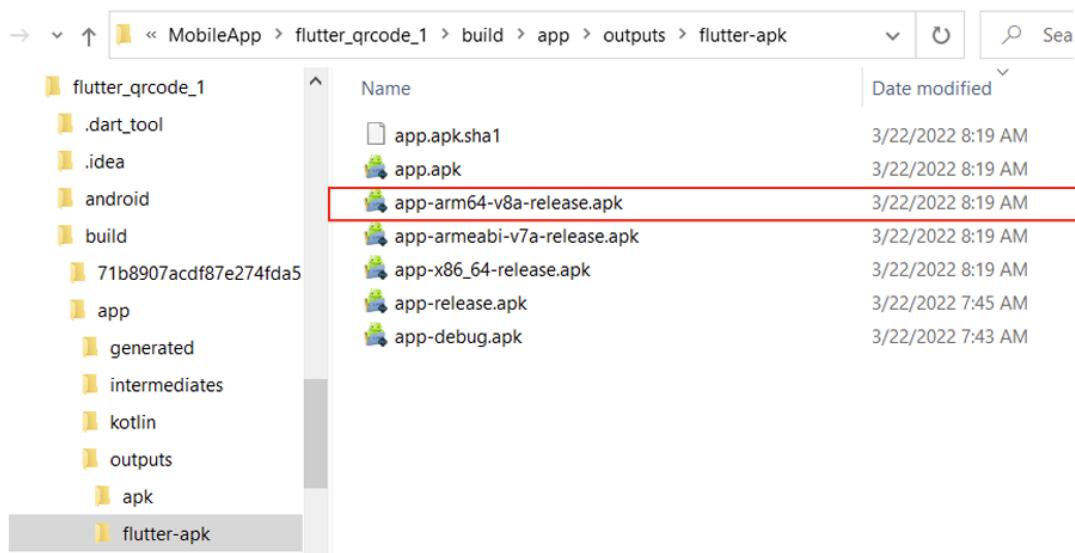


หากต้องการสร้างไฟล์ที่เหมาะสมสำหรับติดตั้งในระบบ 32-bits และ 64-bits

ให้ใช้คำสั่ง flutter build apk --split-per-abi

```
PS D:\2564-2\060233209 Mobile App Dev\MobileApp\flutter_qrcode_1> flutter build apk --split-per-abi
Building with sound null safety
Parameter format not correct -
Running Gradle task 'assembleRelease'...
42.8s
✓ Built build\app\outputs\flutter-apk\app-armeabi-v7a-release.apk (6.8MB).
PS D:\2564-2\060233209 Mobile App Dev\MobileApp\flutter_qrcode_1>
```

Debug my code + packages Dart DevTools Go Live Pi



** จากนั้นนำ apk ไปติดตั้งที่ smartphone android และทดสอบการใช้งาน