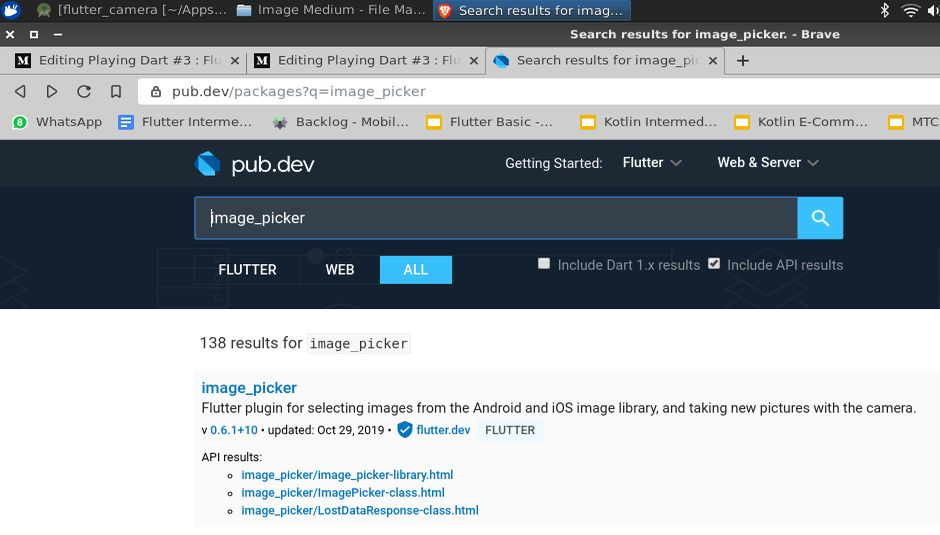
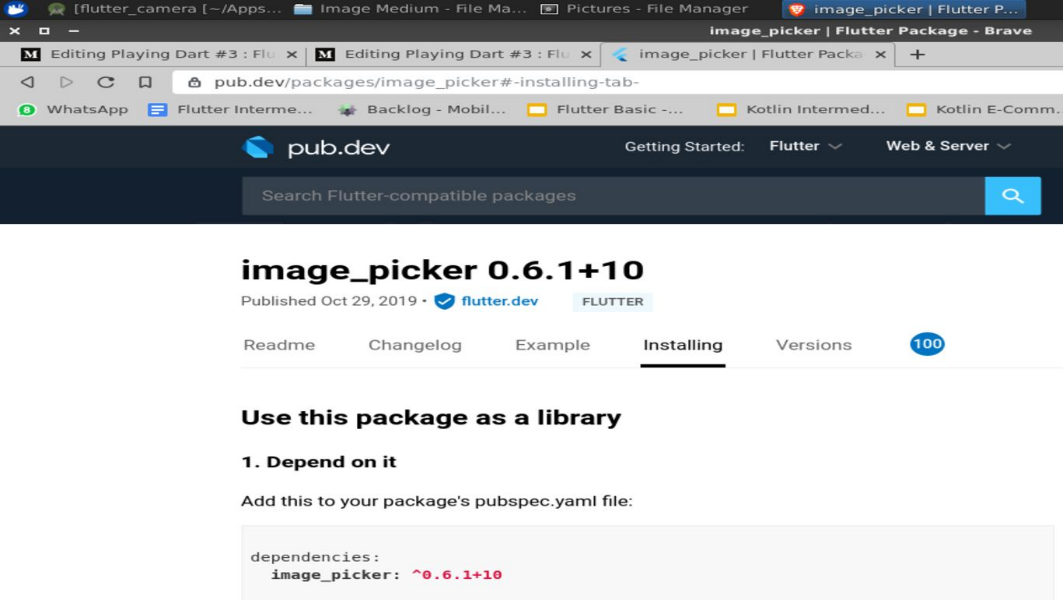
**16. Membuat Fitur Kamera**

Pada modul kali ini kita akan membahas membuat aplikasi yang mana akan meminta akses menggunakan camera smartphone kita dan ketika kita mengambil gambar maka gambar akan ditampilkan dihalaman home, oke bagaimana cara nya langsung saja kita mulai

* Pertama kita buat project baru terlebih dahulu dengan nama flutter\_camera setelah terbuat kita tunggu sampai proses indexing nya selesai
* Sambil menunggu proses indexing kita buka situs pub.dev untuk mendapatkan package tambahan yang bernama image\_picker, kita buka situs tersebut lalu kita masukan kata kunci image\_picker lalu klik tombol search seperti gambar dibawah ini



* Lalu kita pilih yang pertama kali muncul kemudian kita lihat pada bagian Installing kemudian kita copy kan pada bagian dependencies tepatnya di**image\_picker: ^version** ​ untuk ditempatkan di dalam file pubspec.yamltepatnya pada bagian dependenciesnya juga



* Setelah proses indexing file selesai kita menuju pada file pubspec.yaml kita copy kan image\_picker tadi pada bagian dependenciesnya maka kurang lebih seperti ini Kemudian kita klik packgae get :

dependencies:

flutter:

sdk:​ ​ flutter

​ *# The following adds the Cupertino Icons font to your application.*

​ *# Use with the CupertinoIcons class for iOS style icons.*

cupertino\_icons:​ ​ ^0.1.2

image\_picker:​ ​ ^0.6.1+8

* Setelah proses indexing kedua selesai kita menuju pada bagian file main.dart kemudian kita edit file tersebut dengan mengetikan code berikut ini :

import​ ​ 'dart:io'​ ;

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

import​ ​ 'package:image\_picker/image\_picker.dart'​ ;

void​ main() => runApp(MaterialApp(

home: AksesKamera(),

debugShowCheckedModeBanner: ​ false​ ,

));

class​ ​ AksesKamera​ ​ extends​ ​ StatefulWidget​ {

​ @override

\_AksesKameraState createState() => \_AksesKameraState();

}

class​ ​ \_AksesKameraState​ ​ extends​ ​ State​ <​ AksesKamera​ > {

File image;

*//akses kamera*

aksesCamera() ​ async​ {

​ print​ (​ 'Picker is Called'​ );

File img = ​ await​ ImagePicker.pickImage(source: ImageSource.camera);

​ if​ (img != ​ null​ ) {

image = img;

setState(() {});

}

}

​ @override

Widget build(BuildContext context) {

​ return​ Scaffold(

appBar: AppBar(

title: Text(​ 'Akes Kamera'​ ),

backgroundColor: Colors.green,

),

body: Center(

child: Container(

child: image == ​ null​ ? Text(​ 'No Image To Show'​ ) :

Image.file(image),

),

),

floatingActionButton: FloatingActionButton(

backgroundColor: Colors.green,

child: Icon(Icons.add\_a\_photo),

onPressed: aksesCamera,

),

);

}

}

* Setelah selesai mengedit pasti akan ada file yang merah yaitu tepatnya pada file widget\_test.dart, pada file itu kita edit dan kita ganti pada ​ **Home** menjadi **AksesKamera** ​ maka kurang lebih seperti ini full script nya:

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

import​ ​ 'package:flutter\_test/flutter\_test.dart'​ ;

import​ ​ 'package:flutter\_camera/main.dart'​ ;

void​ main() {

testWidgets(​ 'Counter increments smoke test'​ , (WidgetTester tester)

async​ {

*// Build our app and trigger a frame.*

​ await​ tester.pumpWidget(AksesKamera());

​ *// Verify that our counter starts at 0.*

expect(find.text(​ '0'​ ), findsOneWidget);

expect(find.text(​ '1'​ ), findsNothing);

​ *// Tap the '+' icon and trigger a frame.*

​ await​ tester.tap(find.byIcon(Icons.add));

​ await​ tester.pump();

​ *// Verify that our counter has incremented.*

expect(find.text(​ '0'​ ), findsNothing);

expect(find.text(​ '1'​ ), findsOneWidget);

});

}

* Agar dapat berjalan lancar pada saat kita running aplikasi kita didalam iOS kita perlu menambahkan script dibawah ini pada file ​ **info.plist** ​ tepatnya didalam folder ​ **ios**​ /​ **Runner/info.plist** ​ kita ketikan script berikut ini :

....

<​ key​ >NSPhotoLibraryUsageDescription</​ key​ >

<​ string​ >We need this ​ key​ ​ to​ ​ get​ photo ​ from​ gallery </​ string​ >

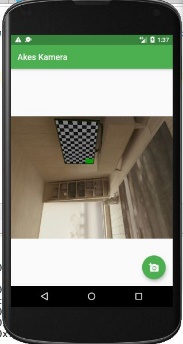
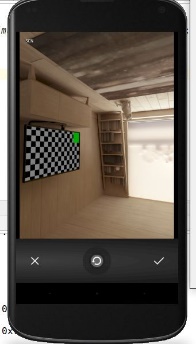
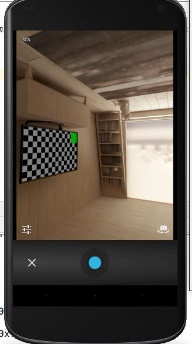
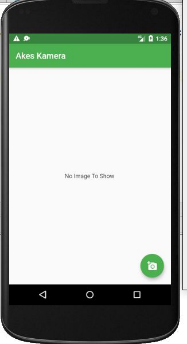
<​ key​ >NSCameraUsageDescription</​ key​ >

<​ string​ >We need this ​ key​ ​ to​ ​ get​ photo ​ from​ camera source </​ string​ >

<​ key​ >NSMicrophoneUsageDescription</​ key​ >

....

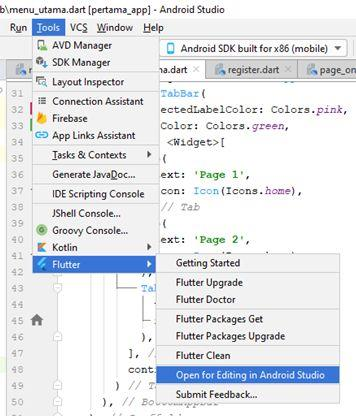
* Kemudian coba kita running aplikasi kita tadi dengan menggunakan emulator ataupun smartphone langsung maka hasilnya kurang lebih akan seperti ini :



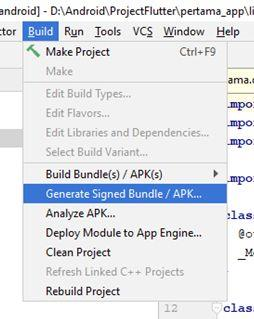
**15. MEMBUAT APK DARI PROJECT FLUTTER**

Adapun langkah kerja yang dilakukan untuk membuat apk dari project flutter, yaitu sebagai berikut:

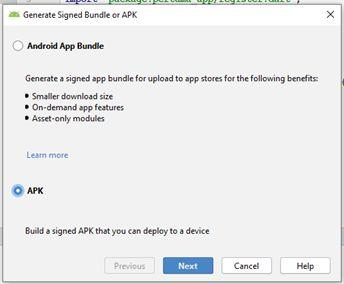
1. Buka project flutter di android studio, klik **Tools** kemudian pilih **flutter** dan pilih **Open For Editing In Android Studio.**



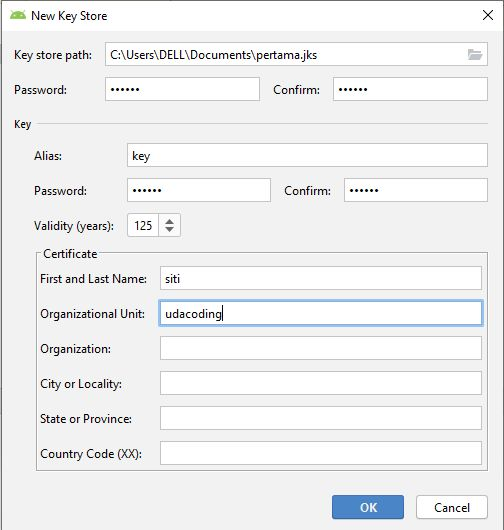
2. Pilih Build dan pilih **Generate Signed Bundle/APK**



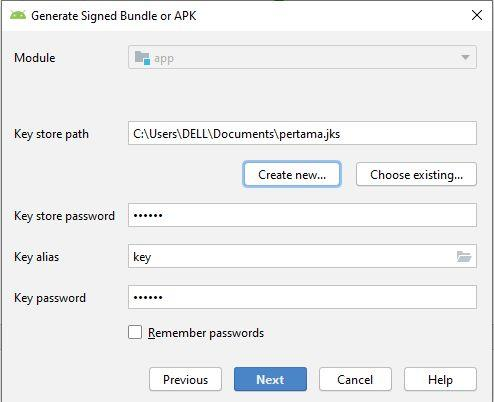
3. Selanjutnya pilih **APK**



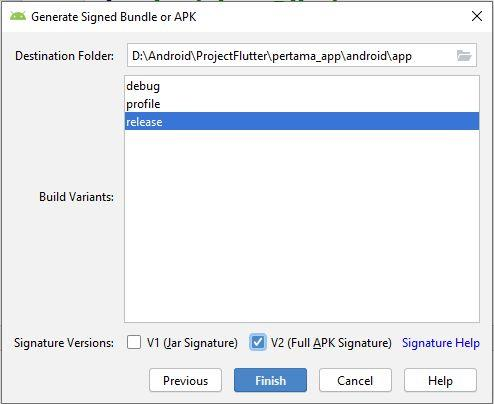
4. Isi form dengan benar



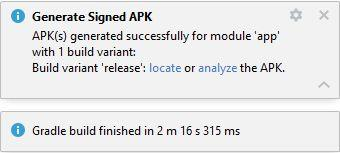
5. Selanjutnya klik OK maka akan muncul tampil seperti berikut ini:



6. Klik Next. Selanjutnya pilih **Release**, hal ini bertujuan untuk release APK yang akan diupload pada playstore. dan klik salah satu Signature Versionnya.



7. Selanjutnya pilih locate untuk melihat APK release



8. Berikut adalah APK sudah release

