

Introduction to Cross Platform Development using Flutter



Mr.Krisada Vivek (Mo)

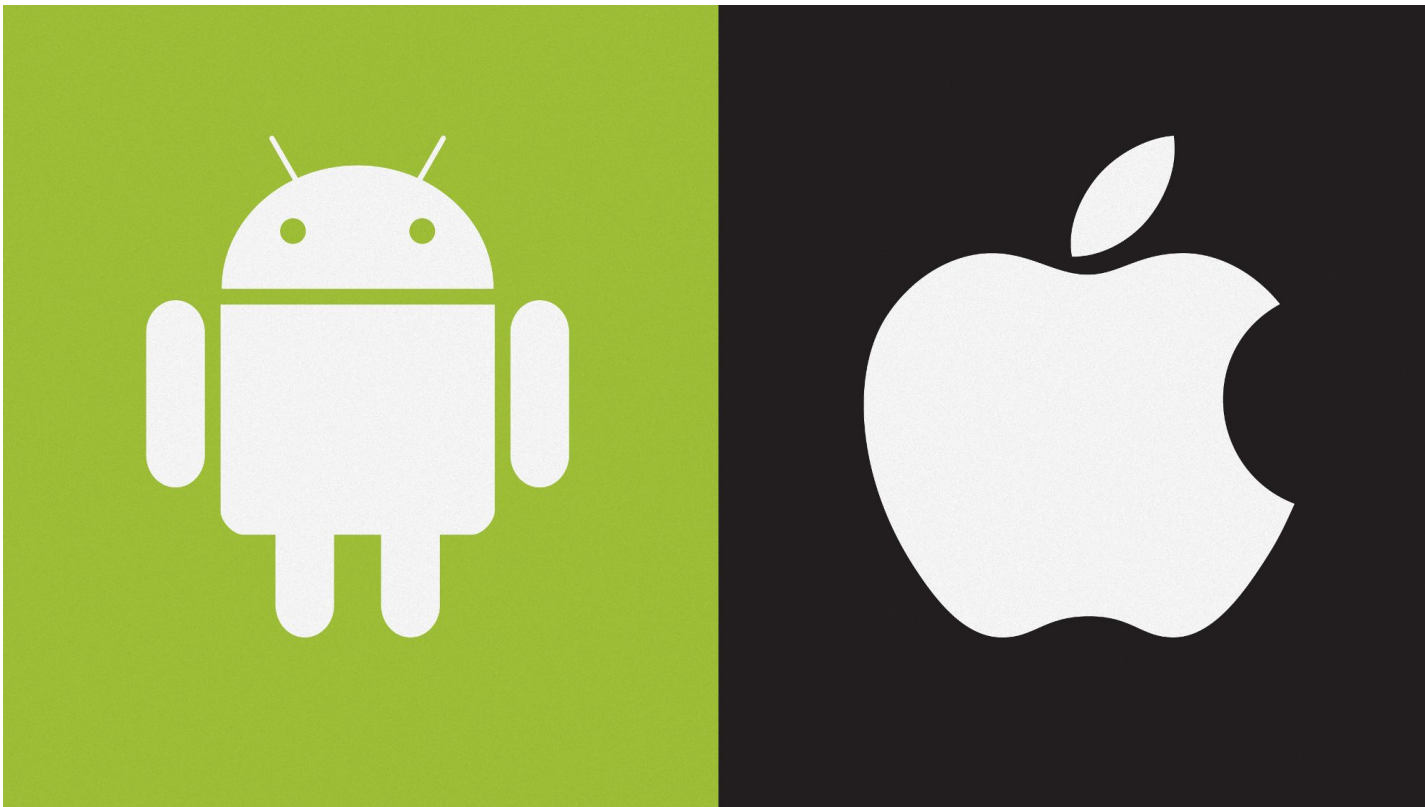
Course Outline (day 1)

- Introduction to flutter past - present
- Set up Flutter SDK
- Flutter's Development Tools (IDE, Simulator)
- Introduction to Dart Programming
- Basic Widget (Text, Image, Container, Column, Row)
- Layout
- Navigation (push, pop)
- State Management (setState)

Chapter 1: Introduction to Flutter



Mobile

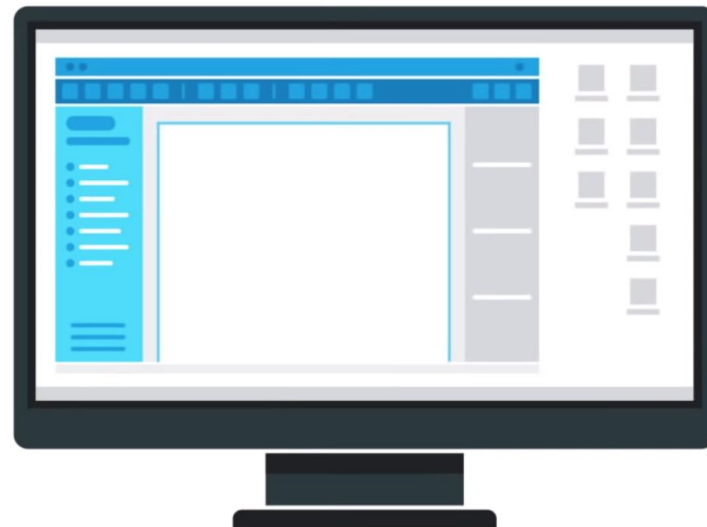
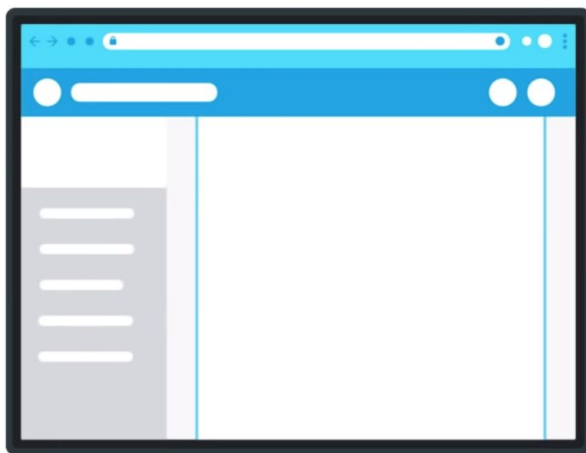
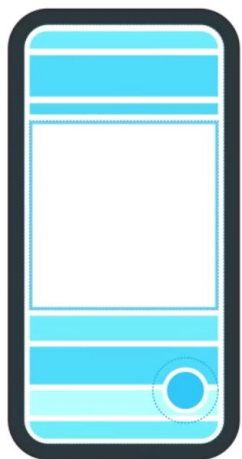


Proverb - สุภาษิต

“ยิงปืนนัดเดียวได้นกสองตัว”

Flutter is Google's UI toolkit for building beautiful, natively compiled applications for [mobile](#), [web](#), and [desktop](#) from a single codebase.

เขียน Flutter ได้ สามารถสร้าง Product รันได้มากถึง 3 Platform*



Beautiful

Productive



Flutter

Fast

Open

Beautiful

No compromises for
your designers

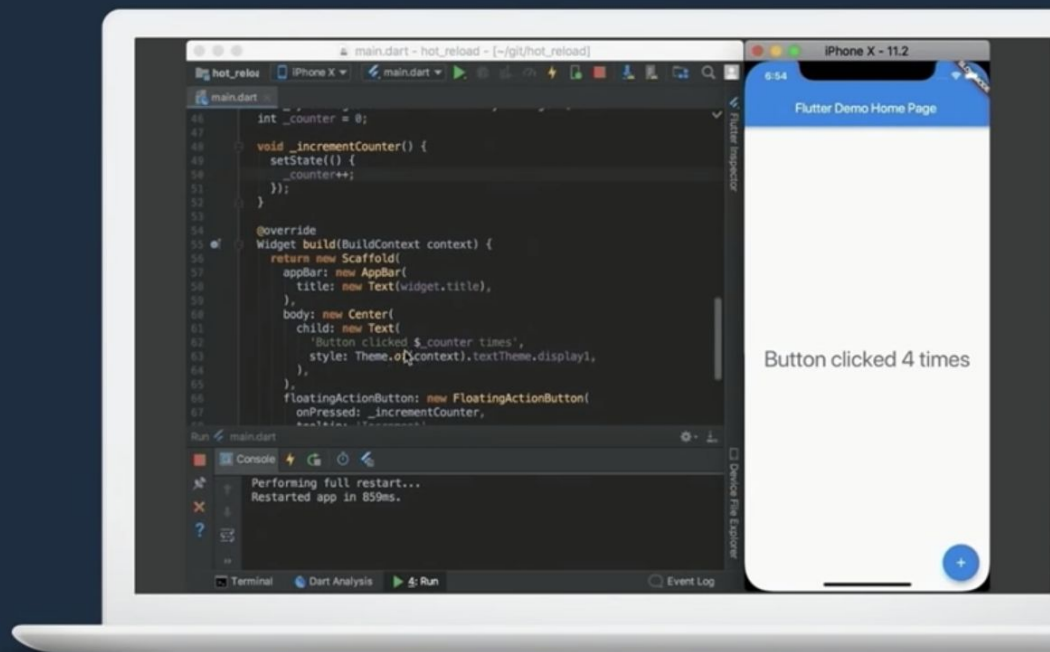


Fast

Take the speed limiter
off your app

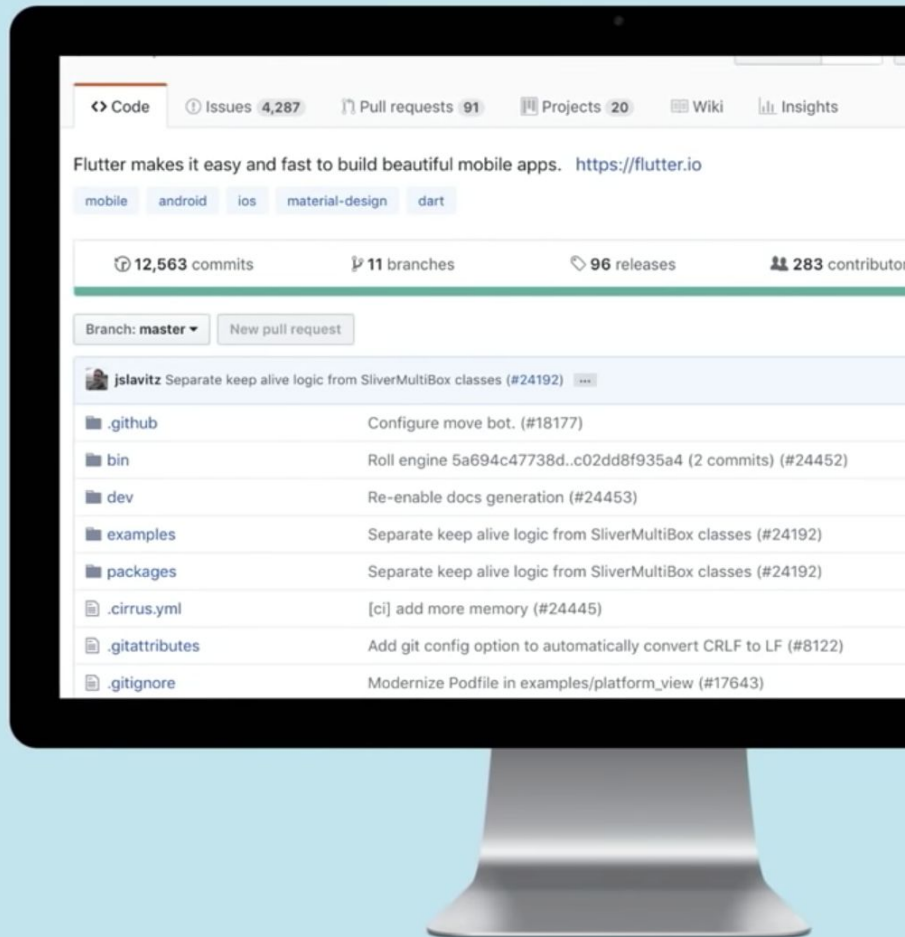


Productive
Develop while your
app is running



Open

Everything is free
and open source





Fast Development

Paint your app to life in milliseconds with Stateful Hot Reload. Use a rich set of fully-customizable widgets to build native interfaces in minutes.



Expressive and Flexible UI

Quickly ship features with a focus on native end-user experiences. Layered architecture allows for full customization, which results in incredibly fast rendering and expressive and flexible designs.



Native Performance

Flutter's widgets incorporate all critical platform differences such as scrolling, navigation, icons and fonts, and your Flutter code is compiled to native ARM machine code using [Dart's native compilers](#).

Who's using Flutter?

Google



ebay

From Native-Mobile Developer Perspective

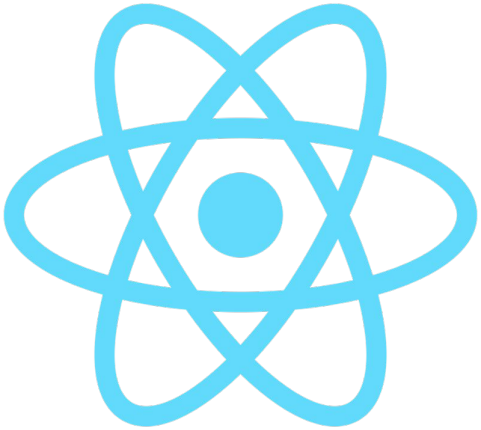
Both Xcode* and Android Studio are fully support for flutter (build and deploy)



**Android
Studio**

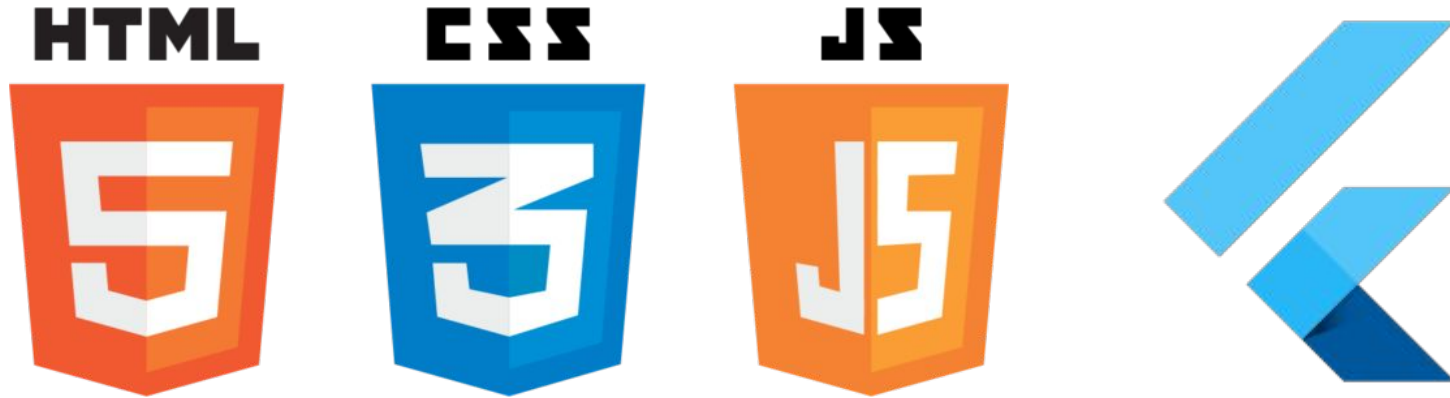
From React Native - JS Developer Perspective

Syntaxing, Code Structure (Concept Component, State Management etc)
similarities



From Web Developer Perspective

Maps HTML/CSS code snippets to Flutter/Dart code equivalents.



HTML/CSS

```
.greybox {  
  background-color: #e0e0e0; /* grey 300 */  
  width: 320px;  
  height: 240px;  
  font: 900 24px Georgia;  
}
```

```
.greybox {  
  
  ....  
  
  font-weight: bold;  
  
  font-family: Georgia;  
  
  font-size: 24px;  
  
  ....  
  
}
```

Flutter

```
var container = new Container( // grey box  
  child: new Text(  
    "Lorem ipsum",  
    style: new TextStyle(  
      fontSize: 24.0  
      fontWeight: FontWeight.w900,  
      fontFamily: "Georgia",  
    ),  
  ),  
  width: 320.0,  
  height: 240.0,  
  color: Colors.grey[300],  
);
```

```
<div class="greybox">  
  
  Lorem ipsum  
  
</div>
```



Chapter 2: Installation and Setup Flutter SDK

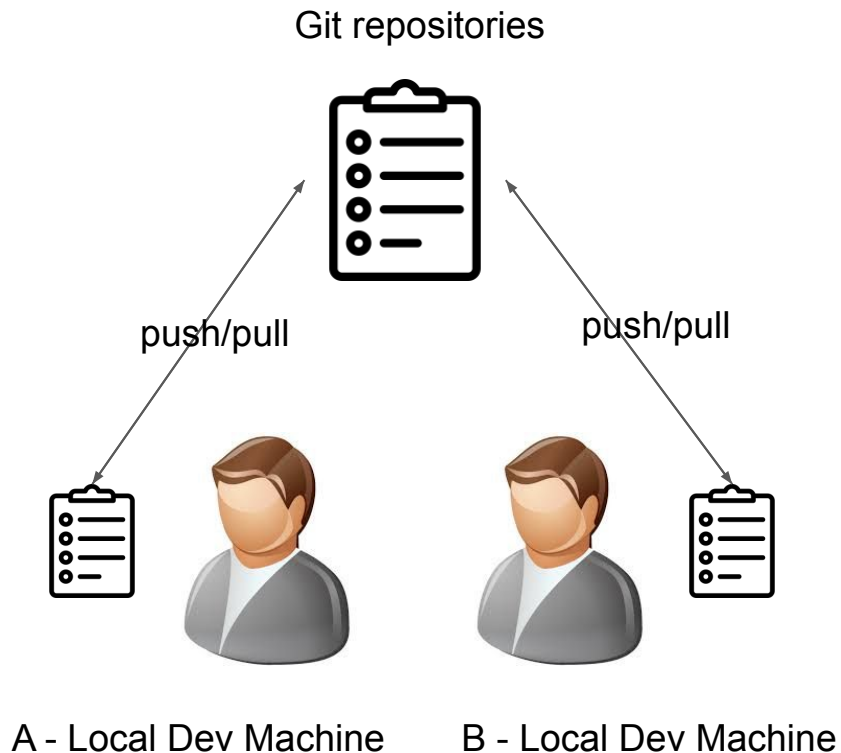
How do i start?

<https://flutter.dev/>

Additional Install

- Git - <https://git-scm.com>
- Git Account- <https://www.github.com>
- Command line - Terminal, Powershell
- Dartlang SDK

Git Concept/ Terminology



Repositories

Shared Code

Collaboration Tools

Working with Teams
(up to hundreds, thousand)

Origin - Local -
Master - Branches

Git commands

git clone <git repo>

git push <from> <to>

git pull <from> <to>

git add . (specific / all files)

git commit -m "any message"

git init -> start git tracking

git config --global user.name ""

git config --global user.email ""

POSTMAN

- Application Platform Interface (API)
- **RESTful** (get, post, put, delete)
 - get - Request Data,
 - post - Create Data,
 - put - Update Data,
 - delete - Remove Data,



<https://www.getpostman.com>
for more info/downloads

Official Support Editor

Note: additional
extension are
require.



Android
Studio

Chapter 3: Installation and Setup Flutter SDK

<Switch to screen>

Chapter 4: Introduction to Dartlang



Dart - Programming language

- Made by Google
- use in Flutter as main develop language.
- Visited <https://dart.dev> for more detail.
- Install on your machine
 - brew install dart (macOS)
 - choco install dart-sdk (windows)



Dart

Chocolatey

The package manager for windows

Install ->

<https://www.chocolatey.org/install>



Further Study

- <https://flutter.dev>
- <https://dart.dev>

<Break for 15 min>

Recap

Introduction to Flutter -> <https://www.flutter.dev>

Install flutter on macOS and Windows, etc.

Setup Android Studio -> Tools for develop mobile application base on android operating system using java or kotlin, from Google,

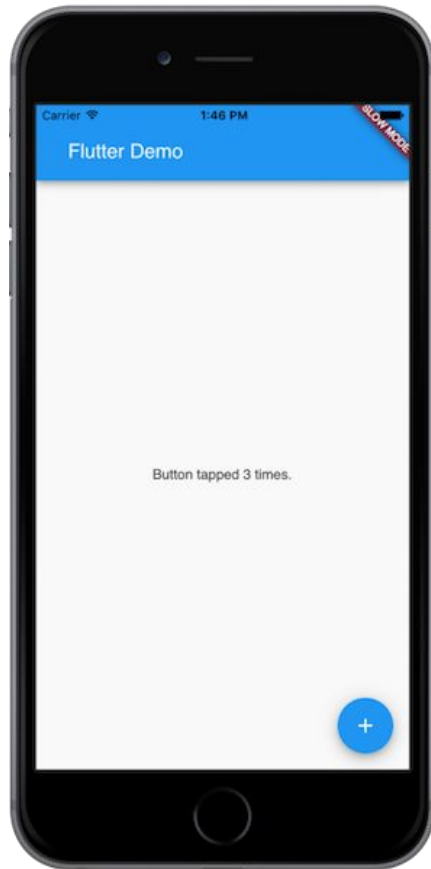
Setup Vscode -> Universal Code editor, from Microsoft

Generating an example flutter application

- Assume that you are already install VScode Android Studio, etc
- Open VScode
- Press ctrl + Shift + p
- Type flutter
- Select options -> flutter: new project
- Let the flutter create process to complete

Flutter Example app

Counter Example App -> by clicking the add sign button at the bottom right corner (called Floating Actions Button) of the bottom right of the screen, the number will update by tab counts and display as text in the middle of the screen



Commands

flutter doctor -v

```
[✓] Flutter (Channel unknown, v1.5.8, on Mac OS X 10.14.6 18G103, locale en)
    • Flutter version 1.5.8 at /Users/Krseisenh/flutter
    • Framework revision 0ba67226ee (7 months ago), 2019-04-24 17:18:28 -0700
    • Engine revision c63d1cf9c9
    • Dart version 2.3.0 (build 2.3.0-dev.0.1 f1f592edce)
```

This sections tell you which Flutter and Dart sdk version you have installed in your machine.

```
[!] Android toolchain - develop for Android devices (Android SDK version 28.0.3)
    • Android SDK at /Users/Krseisenh/Library/Android/sdk
    • Android NDK location not configured (optional; useful for native profiling support)
    • Platform android-28, build-tools 28.0.3
    • Java binary at: /Applications/Android Studio.app/Contents/jre/jdk/Contents/Home/bin/java
    • Java version OpenJDK Runtime Environment (build 1.8.0_202-release-1483-b49-5587405)
    ! Some Android licenses not accepted. To resolve this, run: flutter doctor --android-licenses
```

This sections tell you is Android studio and any android SDK have been install on your machine

```
[!] iOS toolchain - develop for iOS devices (Xcode 10.3)
    • Xcode at /Applications/Xcode.app/Contents/Developer
    • Xcode 10.3, Build version 10G8
    • ios-deploy 1.9.4
    ✗ CocoaPods not installed.
      CocoaPods is used to retrieve the iOS platform side's plugin code that responds to your plugin usage.
      Without resolving iOS dependencies with CocoaPods, plugins will not work on iOS.
      For more info, see https://flutter.dev/platform-plugins
      To install:
        brew install cocoapods
        pod setup
```

This sections tell you is there any mobile device attach and ready to use as deploy device(s)

```
[✓] Android Studio (version 3.5)
    • Android Studio at /Applications/Android Studio.app/Contents
    • Flutter plugin version 39.0.3
    • Dart plugin version 191.8423
    • Java version OpenJDK Runtime Environment (build 1.8.0_202-release-1483-b49-5587405)
```

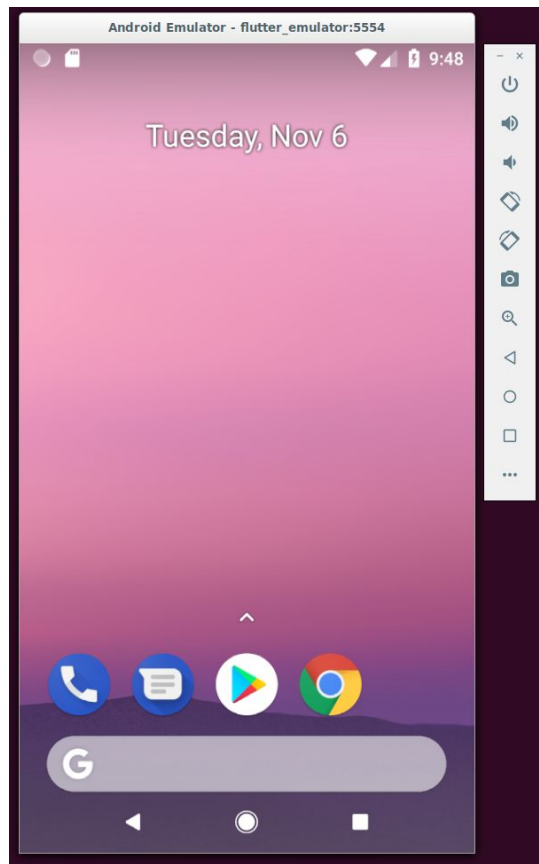
```
[✓] VS Code (version 1.40.0)
    • VS Code at /Applications/Visual Studio Code.app/Contents
    • Flutter extension version 3.6.0
```

This sections tell you is there any mobile device attach and ready to use as deploy device(s)

```
[✓] Connected device (1 available)
    • SM N950F • 988a1b41555751595930 • android-arm64 • Android 9 (API 28)
```


flutter run

- Install you flutter application to any connected device (physical, emulator, simulator)
- open -a Simulator (for mac user who want to deploy in iOS device)
- You can launch Android Emulator from **vscode** (this take awhile for the android emulator to startup, so be patient at this point :))
- Once every this is up, hits “flutter run -d all” in your terminal (powershell)
- And wait for the magic



Other known commands

flutter doctor -v -> this command helps you identify if your computer is ready for develop flutter application.

flutter upgrade -> update flutter SDK to the latest version

r - Hot Reloads -> take an instant update to your flutter application

shift + r - Hot Restart -> restart your flutter application to the initial state

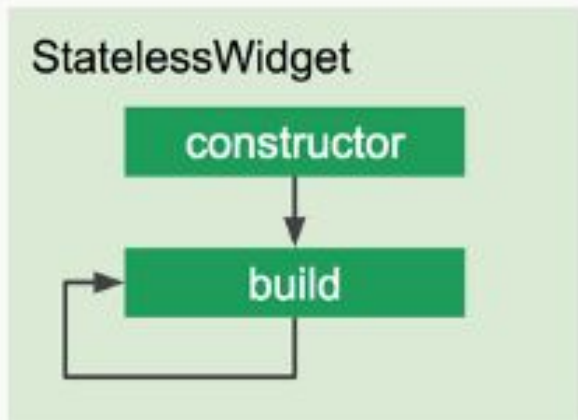
Chapter 5: Widgets

Covid Patient Form

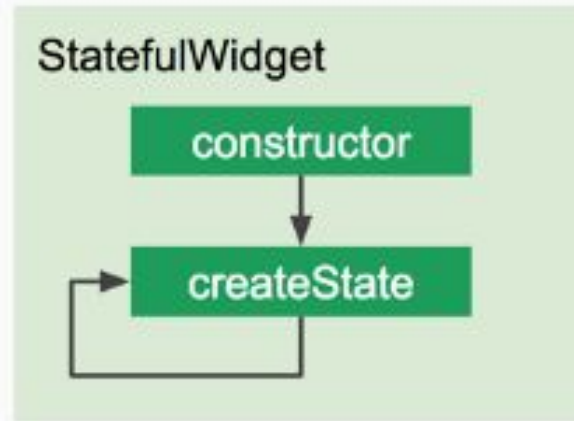
- Objective -> Create a form and pass data between pages.
- Code complexity -> Beginner
- Use Widget
 - Form
 - TextFormField
 - RadioButton
 - CheckBox
 - Navigation between page
- Final code at here -> https://github.com/Krseisenh/covid_form

The screenshot shows a mobile application interface for a 'Patient Form'. The title bar is blue with the text 'Patient Form'. The form consists of several input fields: 'Firstname', 'Lastname', and 'Age' are text input fields. 'Gender' is a radio button group with options 'Male' and 'Female'. 'Symptoms' is a list of three items: 'ไอ' (Cough), 'เจ็บคอ' (Sore throat), and 'มีไข้' (Fever), each followed by a checkbox. A 'Save' button is located at the bottom right of the form. The bottom of the screen shows a standard Android navigation bar with three icons: a home button, a back button, and a recent apps button.

Stateless and Stateful Widgets

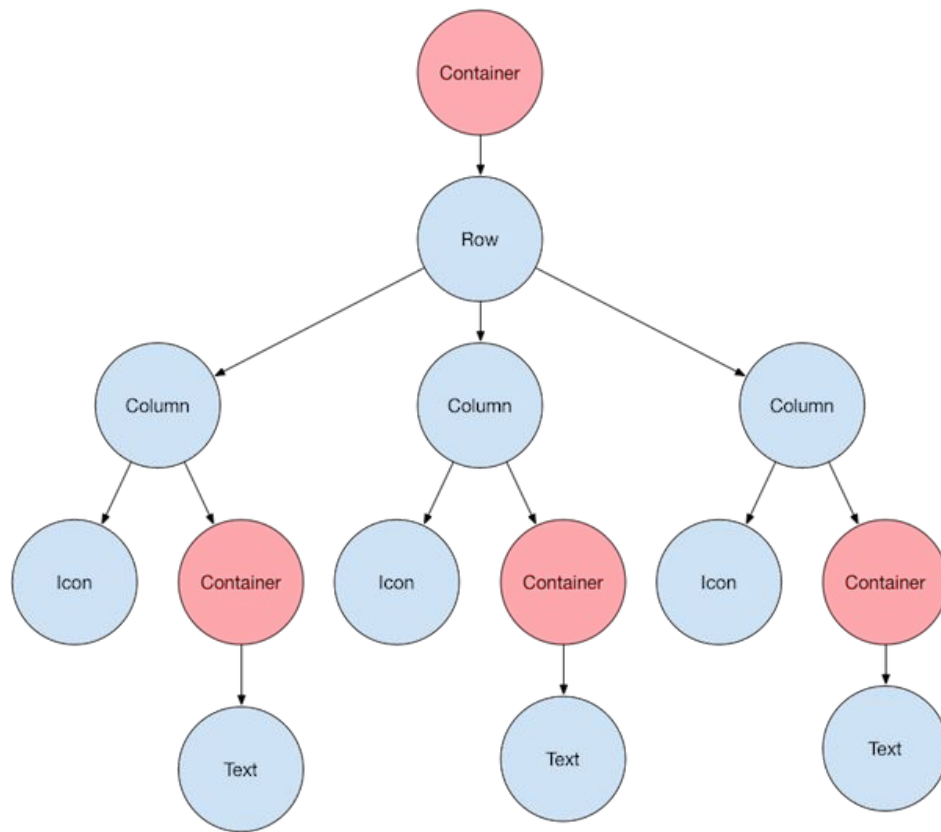


A single StatelessWidget can build in many different BuildContexts



A StatefulWidget creates a new State object for each BuildContext

Widget Tree



Widget: Container

```
Center(  
  child: Container(  
    margin: const EdgeInsets.all(10.0),  
    color: Colors.amber[600],  
    width: 48.0,  
    height: 48.0,  
  ),  
)
```

Widget: Text

```
Text(  
  'Hello, $_name! How are you?',  
  textAlign: TextAlign.center,  
  overflow: TextOverflow.ellipsis,  
  style: TextStyle(fontWeight: FontWeight.bold),  
)
```


Widget: Image

```
Image.asset('images/cat.png')
```

```
Image.network('https://flutter.github.io/assets-for-api-docs/assets/widgets/owl-2.jpg')
```

Widget: Column

```
Column(  
  crossAxisAlignment: CrossAxisAlignment.start,  
  mainAxisAlignment: MainAxisAlignment.min,  
  children: <Widget>[  
    Text('We move under cover and we move as one'),  
    Text('Through the night, we have one shot to live another day'),  
    Text('We cannot let a stray gunshot give us away'),  
    Text('We will fight up close, seize the moment and stay in it'),  
    Text('It's either that or meet the business end of a bayonet'),  
    Text('The code word is 'Rochambeau,' dig me?'),  
    Text('Rochambeau!', style: DefaultTextStyle.of(context).style.apply(fontSizeFactor: 2.0)),  
  ],  
)
```

Widget: Row

```
Row(  
  children: <Widget>[  
    Expanded(  
      child: Text('Deliver features faster', textAlign: TextAlign.center),  
    ),  
    Expanded(  
      child: Text('Craft beautiful UIs', textAlign: TextAlign.center),  
    ),  
    Expanded(  
      child: FittedBox(  
        fit: BoxFit.contain, // otherwise the logo will be tiny  
        child: const FlutterLogo(),  
      ),  
    ),  
  ],  
)
```

Widget: Buttons

```
const RaisedButton(  
  onPressed: null,  
  child: Text(  
    'Disabled Button',  
    style: TextStyle(fontSize: 20)  
  ),  
),
```

```
FlatButton(  
  color: Colors.blue,  
  textColor: Colors.white,  
  disabledColor: Colors.grey,  
  disabledTextColor: Colors.black,  
  padding: EdgeInsets.all(8.0),  
  splashColor: Colors.blueAccent,  
  onPressed: () {  
    /*...*/  
  },  
  child: Text(  
    "Flat Button",  
    style: TextStyle(fontSize: 20.0),  
  ),  
)
```

Widget: ListView

```
ListView(  
  padding: const EdgeInsets.all(8),  
  children: <Widget>[  
    Container(  
      height: 50,  
      color: Colors.amber[600],  
      child: const Center(child: Text('Entry A')),  
    ),  
    Container(  
      height: 50,  
      color: Colors.amber[500],  
      child: const Center(child: Text('Entry B')),  
    ),  
    Container(  
      height: 50,  
      color: Colors.amber[100],  
      child: const Center(child: Text('Entry C')),  
    ),  
  ],  
)
```

Widget: GridView

```
GridView.count(  
  primary: false,  
  padding: const EdgeInsets.all(20),  
  crossAxisSpacing: 10,  
  mainAxisSpacing: 10,  
  crossAxisCount: 2,  
  children: <Widget>[  
    Container(  
      padding: const EdgeInsets.all(8),  
      child: const Text("He'd have you all unravel at the"),  
      color: Colors.teal[100],  
    ),  
    Container(  
      padding: const EdgeInsets.all(8),  
      child: const Text('Heed not the rabble'),  
      color: Colors.teal[200],  
    ),  
  ],  
)
```

Widget: TextFormField

```
TextFormField(  
  decoration: const InputDecoration(  
    icon: Icon(Icons.person),  
    hintText: 'What do people call you?',  
    labelText: 'Name *',  
  ),  
  onSave: (String value) {  
    // This optional block of code can be used to run  
    // code when the user saves the form.  
  },  
  validator: (String value) {  
    return value.contains('@') ? 'Do not use the @ char.' : null;  
  },  
)
```

Widget: Radio Button

```
leading: Radio(  
  value: SingingCharacter.lafayette,  
  groupValue: _character,  
  onChanged: (SingingCharacter value) {  
    setState(() { _character = value; });  
  },  
),
```


Widget: CheckBox

```
Checkbox(  
  value: monVal,  
  onChanged: (bool value) {  
    setState(() {  
      monVal = value;  
    });  
  },  
)
```

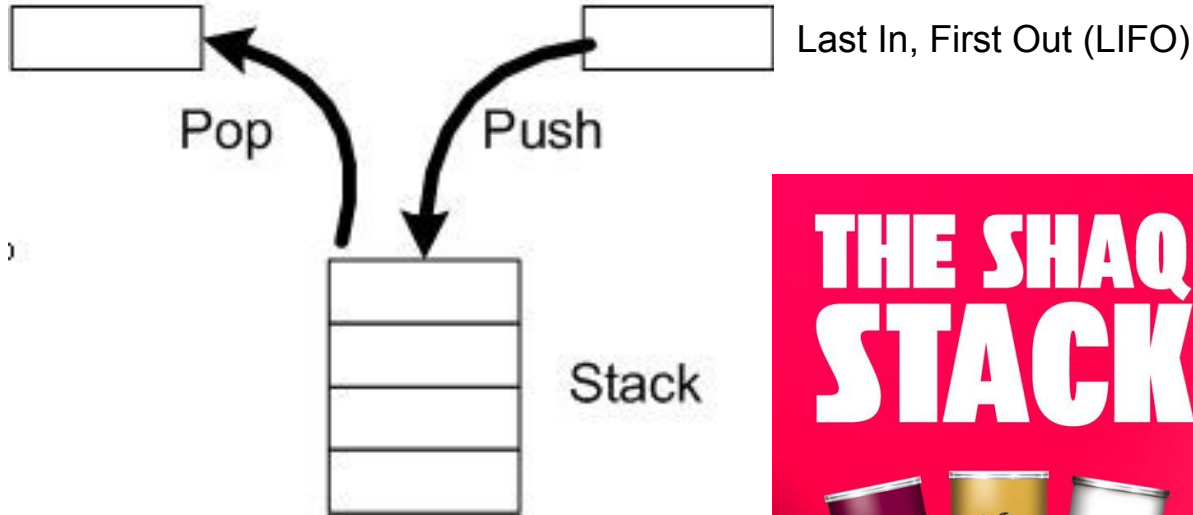
Alignments

Further Study =>

[Flutter Layout Cheat Sheet. Do you need simple layout samples for... | by Tomek Polański | Flutter Community](#)

Navigation

Stack Concept (Push, Pop)



```
// Within the 'FirstRoute' widget
```

```
onPressed: () {
```

```
  Navigator.push(
```

```
    context,
```

```
    MaterialPageRoute(builder: (context) => SecondRoute()),
```

```
  );
```

```
}
```

PUSH

POP

```
// Within the SecondRoute widget
```

```
onPressed: () {
```

```
  Navigator.pop(context);
```

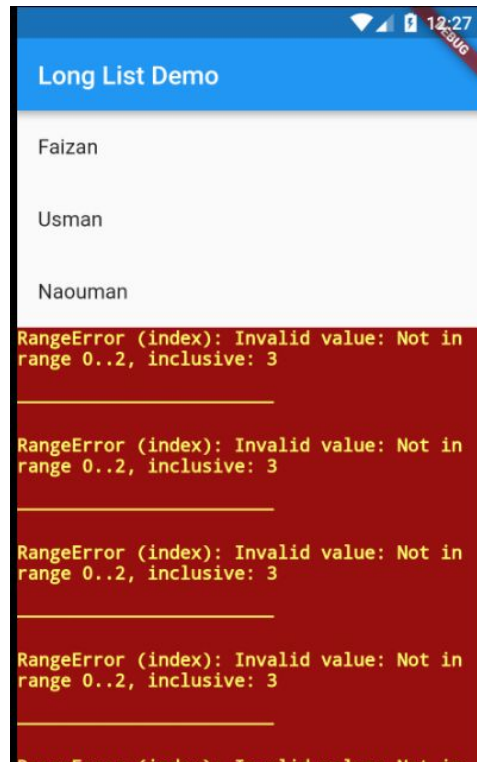
```
}
```

Passing data between pages

1. Create 1st page
2. In the middle of 1st page, create a button
3. Inside the button create function
``Navigator.push(context, MaterialPageRoute(builder: (_)=> SecondPage()))``
4. Create 2nd Page
5. On the 2nd, create a button and named `back`
6. Inside the button create function
``Navigator.pop(context)``

Error in Flutter

This is an errors dialog/screen when you got an errors in your flutter app



Chapter 9: Introduction to State Management

State management -> setState

```
void _incrementCounter() {  
    setState(() {  
        _counter++;  
    });  
}
```

```
Text(  
    '$_counter',  
    style: Theme.of(context).textTheme.headline4,  
),
```

Workshop 1

Different Type of StateManagment

- setState
- Scoped_model -> https://pub.dev/packages/scoped_model
- Bloc -> <https://pub.dev/packages/bloc>
- MobX -> <https://pub.dev/packages/mobx>

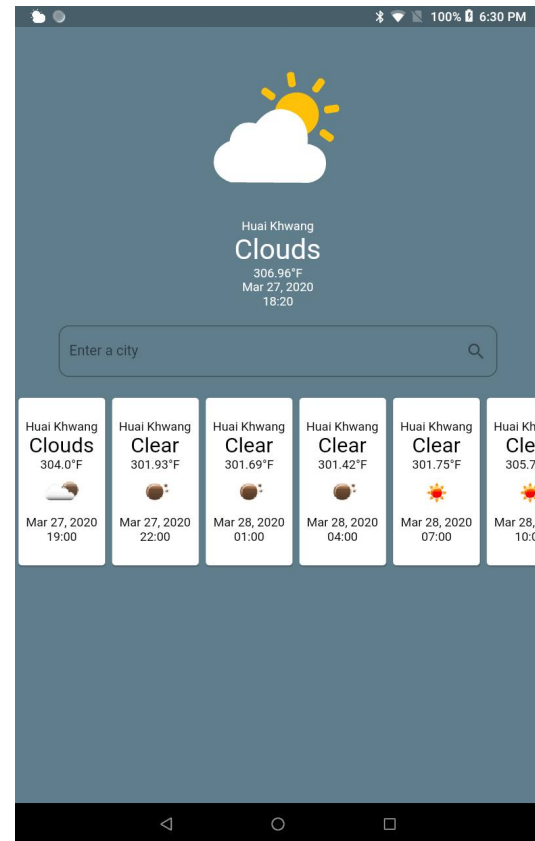
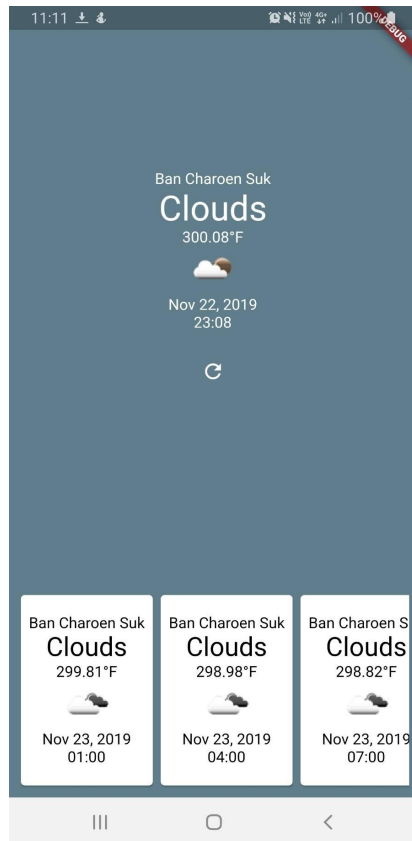
Global Local State Management Concept

- We want to access any data in any page
- Spaghetti Code
- Performance
- Varies update/render interval

“Talk is cheap. Show me the code.”

Linus Torvalds

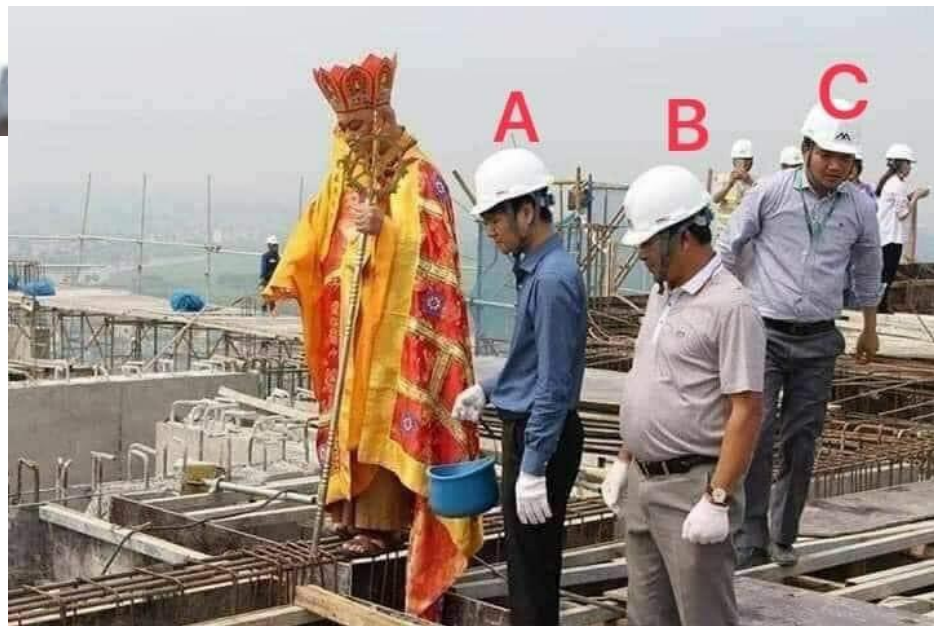
Today Weather App



Course Outline (Day 2)

- Application Architecture
- Working with libs - location, MobX, native_splash_screen, flutter_launcher_icon
- Introduction to Material Design/ Cupertino Design

Application Architecture





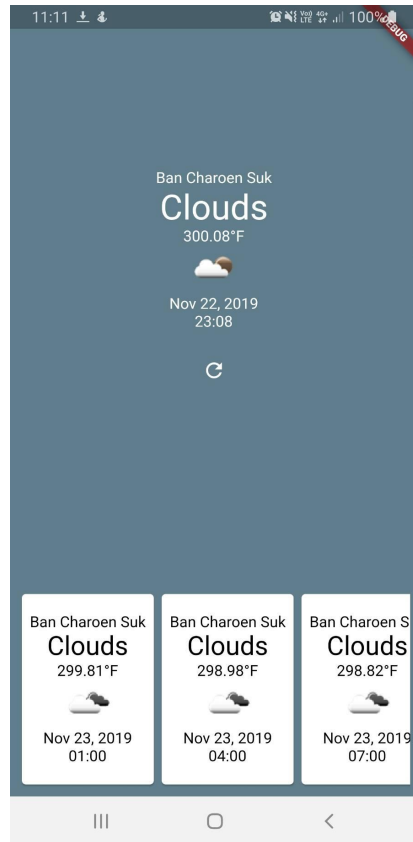
<Discussion Panel>

WorkShop 2 SPA + stateMangement

Today Weather

- Objective -> refactor code, setState .
- Code complexity-> intermediate
- Use Widget
 - Text
 - Image
 - Row
 - Column
- Final code at here ->

https://github.com/3IVv0w/today_weather.git



Today Weather - MobX

- Objective -> refactor code, setState .
- Code complexity-> intermediate - Advanced
- Use Widget
 - Text
 - Image
 - Row
 - Column
 - Animation
- Final code at here -> [Krseisenh/weather_mobx \(github.com\)](https://github.com/Krseisenh/weather_mobx)

Flutter Weather App

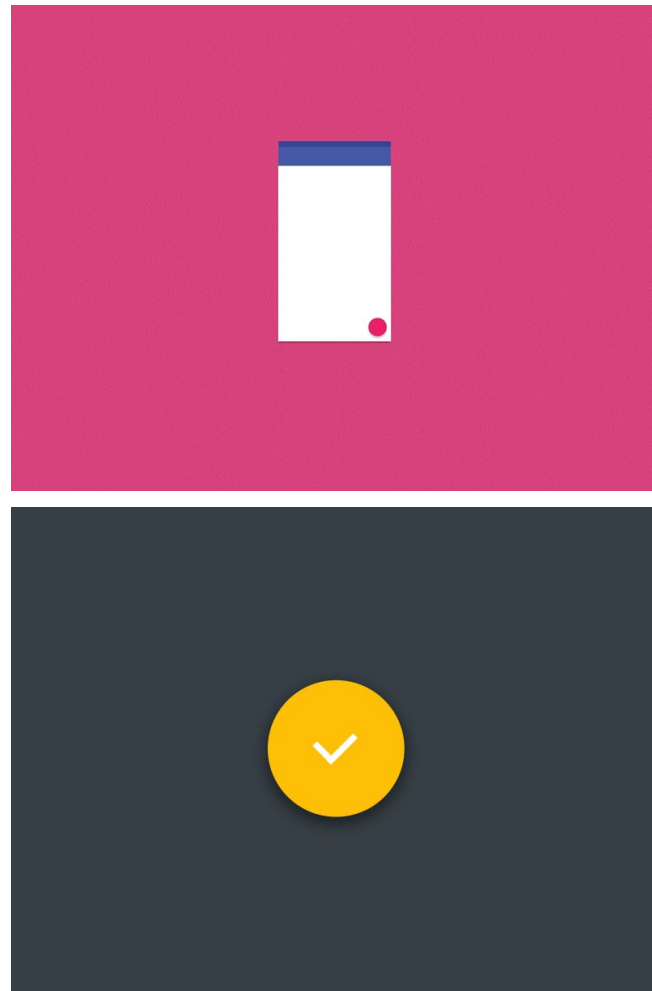
- cd project
- Take a look at the source code
- Create an .env file
- GET an API KEY <https://openweathermap.org/>
- In POSTMAN <https://getpostman.com> ->
`https://api.openweathermap.org/data/2.5/weather?APPID=${apiKey}&lat=12.67893&lon=101.133118`
- Ref. <https://github.com/dragosholban/FlutterWeatherApp>

Introduction to Material Design

Material Design - <https://material.io/>

Design Principle consist of 3 rules

- Material - surface, edge shadows
- Intentional
- Motion



Basic Feature

- Splash Screen - flutter_native_splash
- Login/Register - Dart - (Dio + HTTP), Dart -Firebase Auth
- Setting Page -
- Home

Introduction to Flutter CI/CD

Manual Build an APK/AAB

Commands

- Generated keystore and update gradle file
- flutter clean
- flutter build apk --release
- flutter build appbundle --release

Manual Build an .app/ipa

- Require xcode and developer account

Codemagic

<https://codemagic.io/>