

Android Development

Akshath Jain

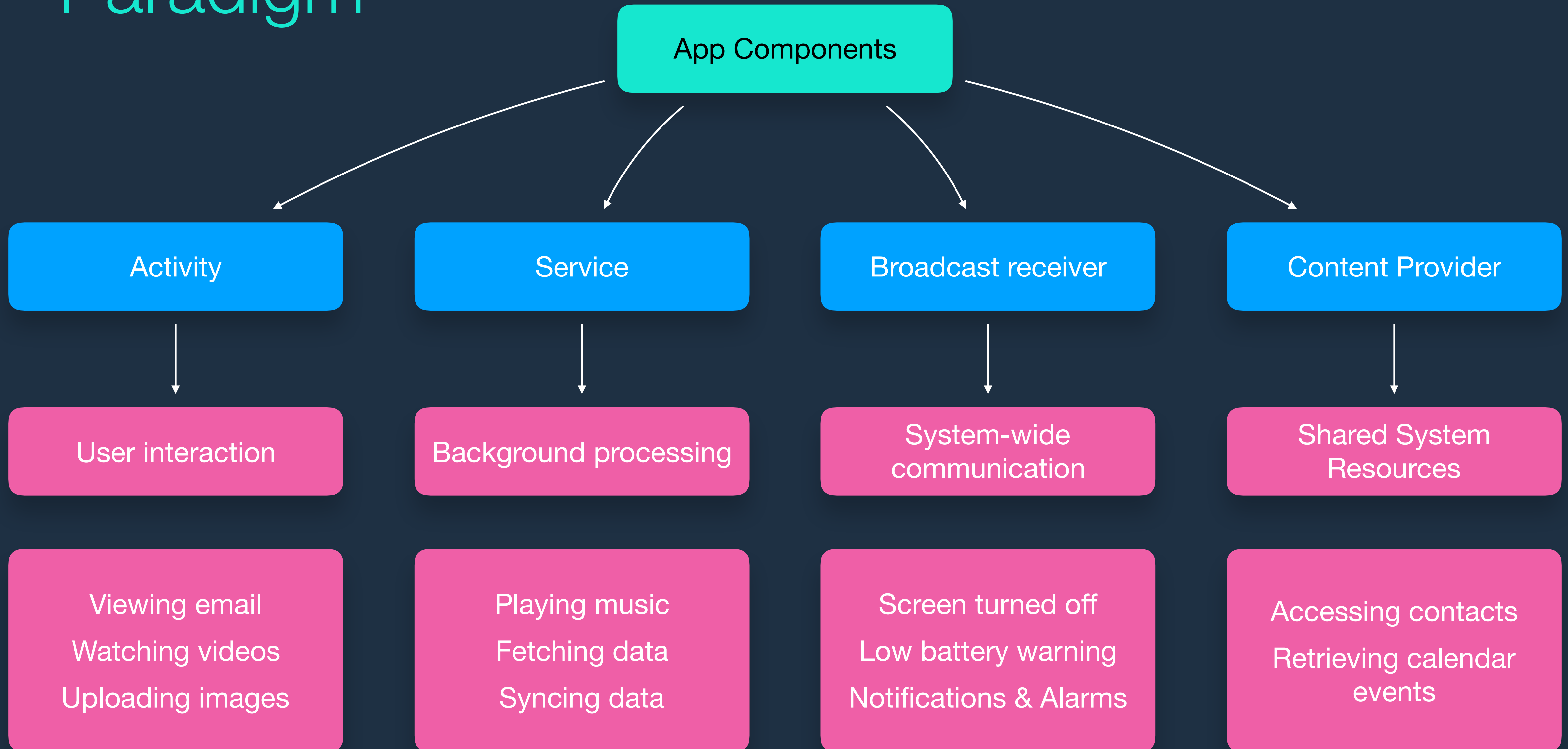
Mobile & IoT — Spring 2019



<https://developer.android.com/studio/>

Paradigm

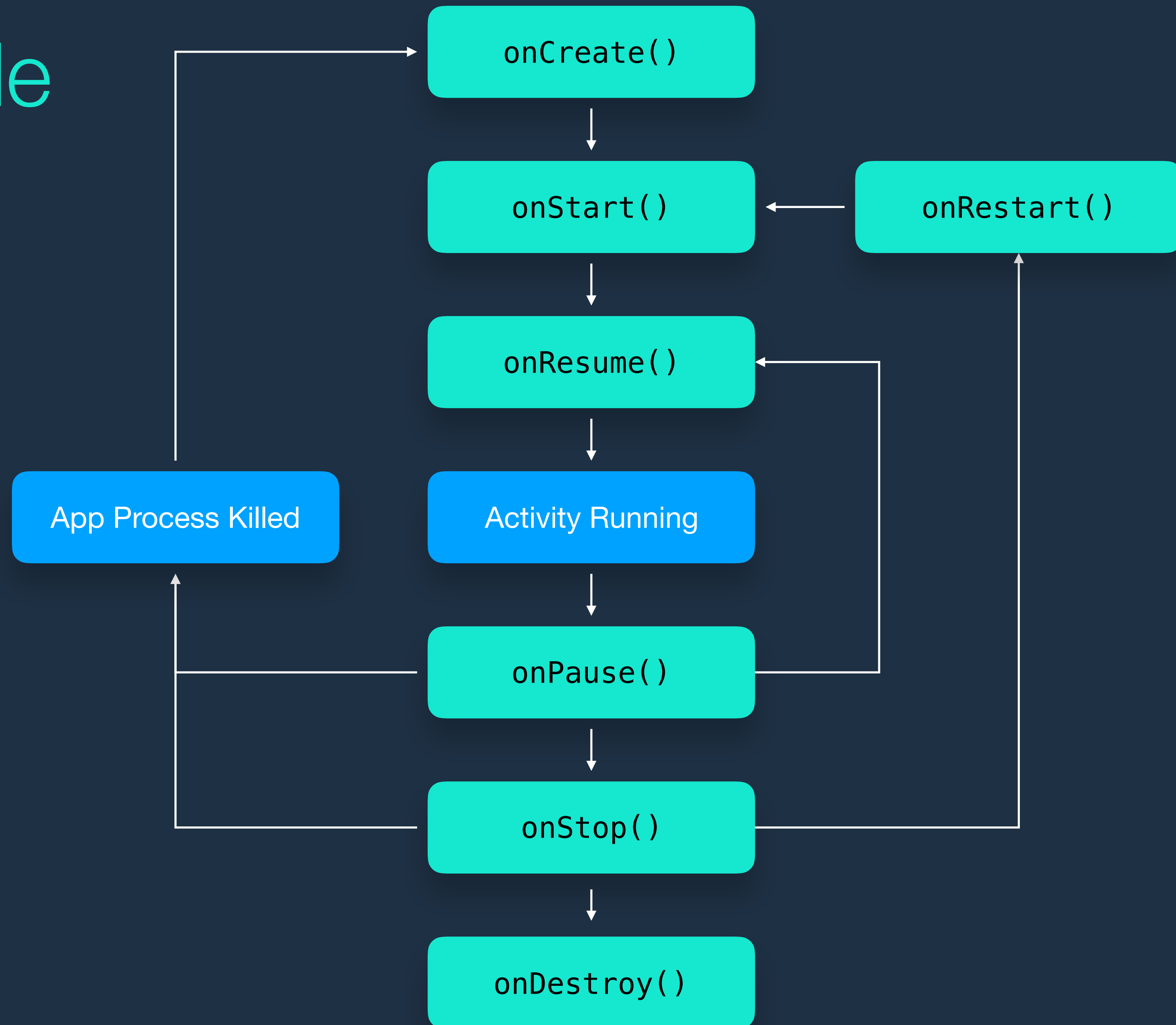
Paradigm



Activity

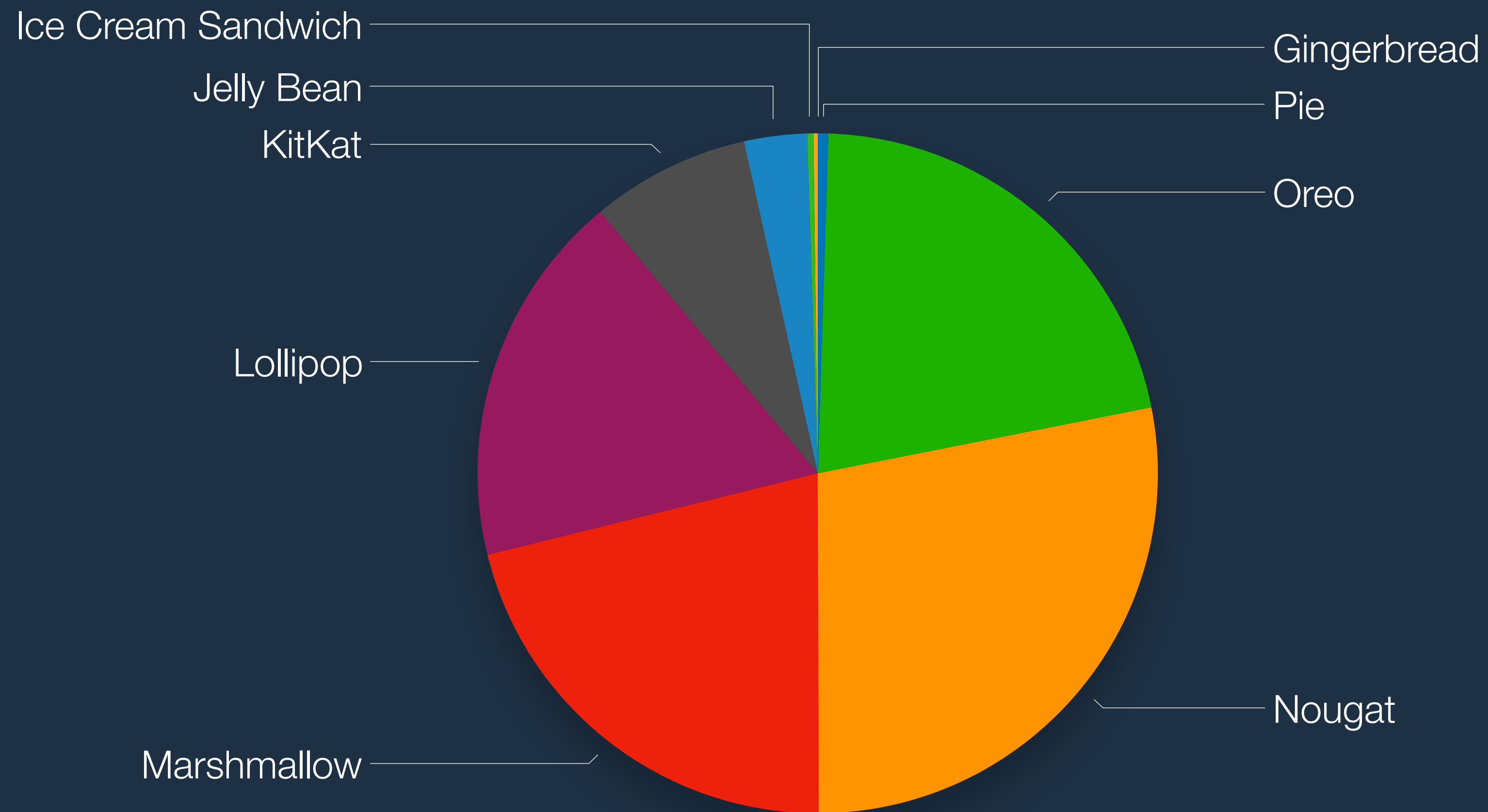
Lifecycle

Lifecycle

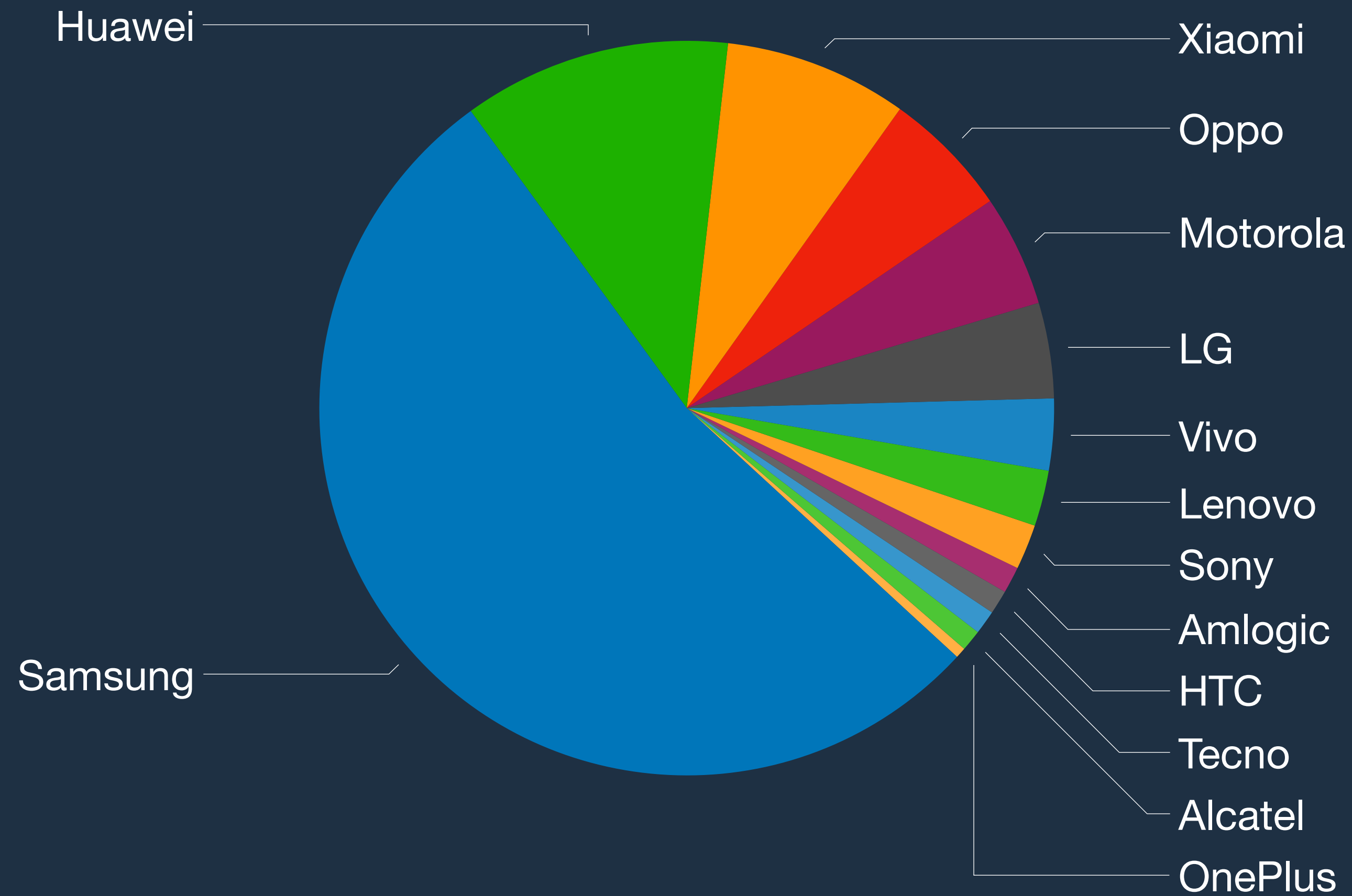


Considerations

Software Fragmentation




Hardware Fragmentation



Hello World

Create New Project



Create Android Project

Application name

AndroidHelloWorld

Company domain

akshathjain.com

Project location

/Users/akshath/Documents/CMU/Spring2019/17781-TA

...

Package name

com.akshathjain.androidhelloworld

Edit

☐ Include C++ support

☐ Include Kotlin support


Cancel

Previous

Next

Finish

Create New Project

 Target Android Devices

Select the form factors and minimum SDK

Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☒ **Phone and Tablet**

API 15: Android 4.0.3 (IceCreamSandwich)▼

By targeting **API 15 and later**, your app will run on approximately **100%** of devices. [Help me choose](#)

☐ Include Android Instant App support

☐ **Wear**

API 21: Android 5.0 (Lollipop)▼

☐ **TV**

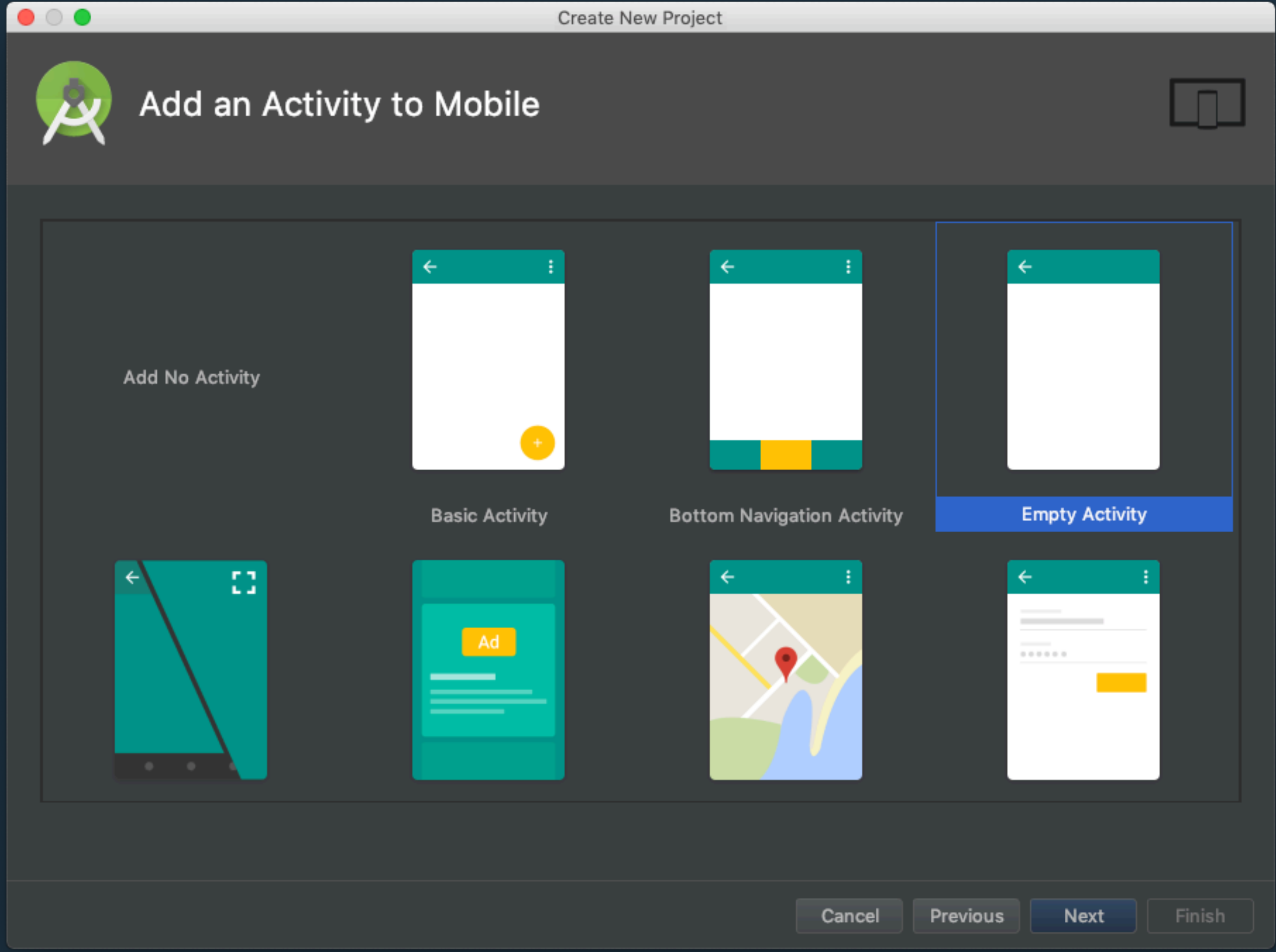
API 21: Android 5.0 (Lollipop)▼

☐ **Android Auto**


☐ **Android Things**

API 24: Android 7.0 (Nougat)▼


CancelPreviousNextFinish




Create New Project



Configure Activity



Creates a new empty activity



Activity Name:

☒ Generate Layout File

Layout Name:

☒ Backwards Compatibility (AppCompat)

The name of the activity class to create

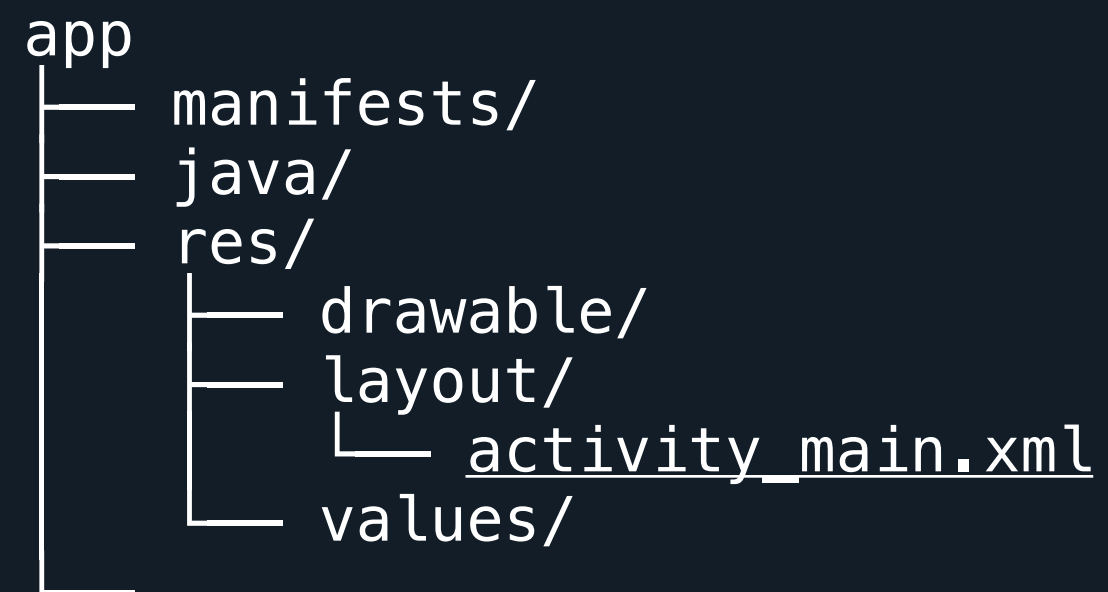
Cancel

Previous

Next

Finish

Code



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true" />

</RelativeLayout>
```


Code

```
app
├── manifests/
├── java/
│   └── MainActivity.java
└── res/
```

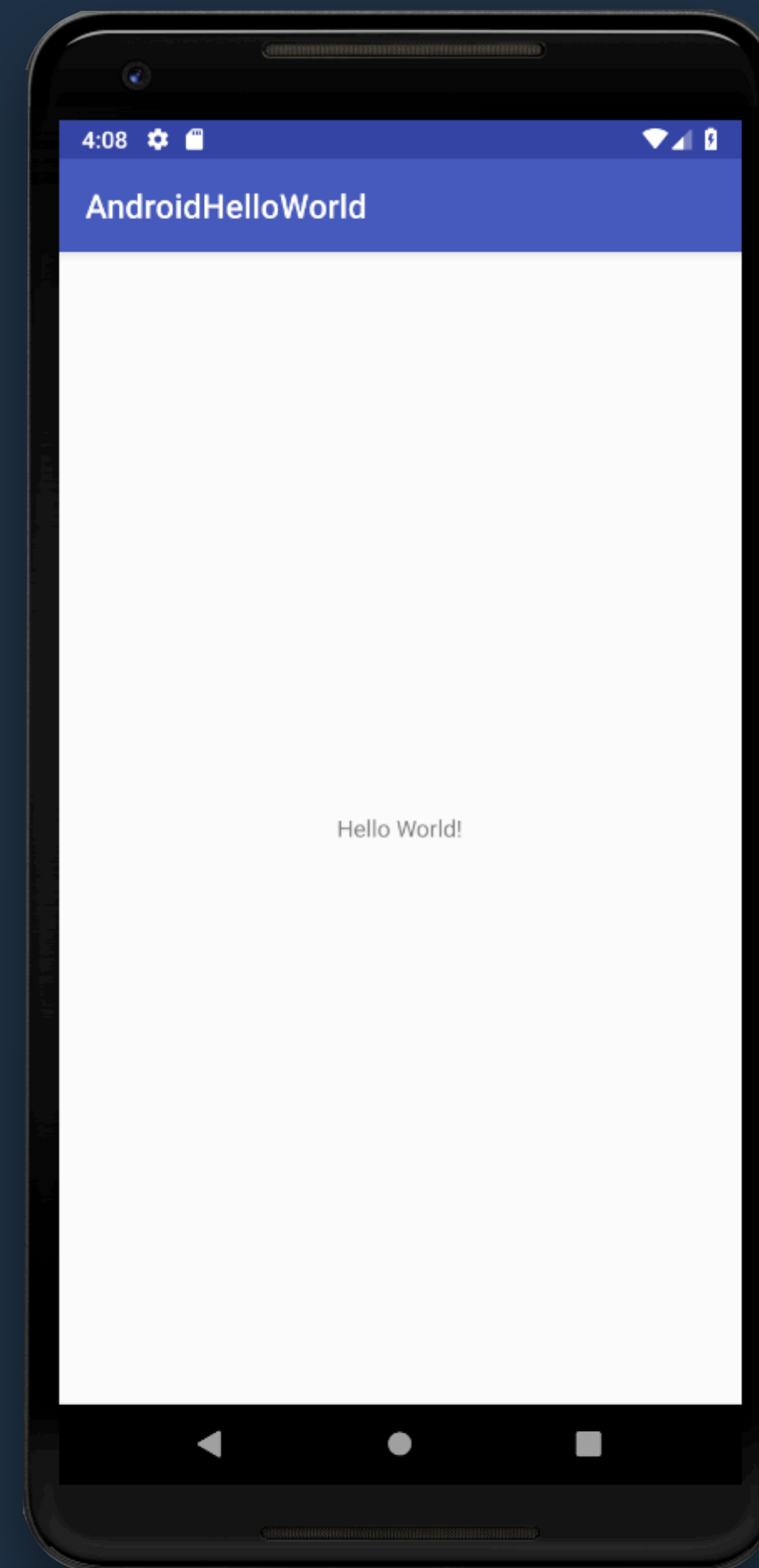
```
package com.akshathjain.androidhelloworld;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

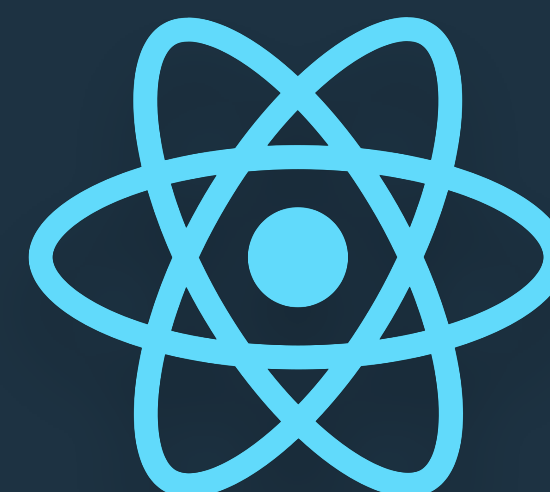
Emulation



Cross Platform Frameworks

Akshath Jain

Mobile & IoT — Spring 2019





Akshath Jain

Mobile & IoT / Crash Course 2019





Fast

Native

Efficient

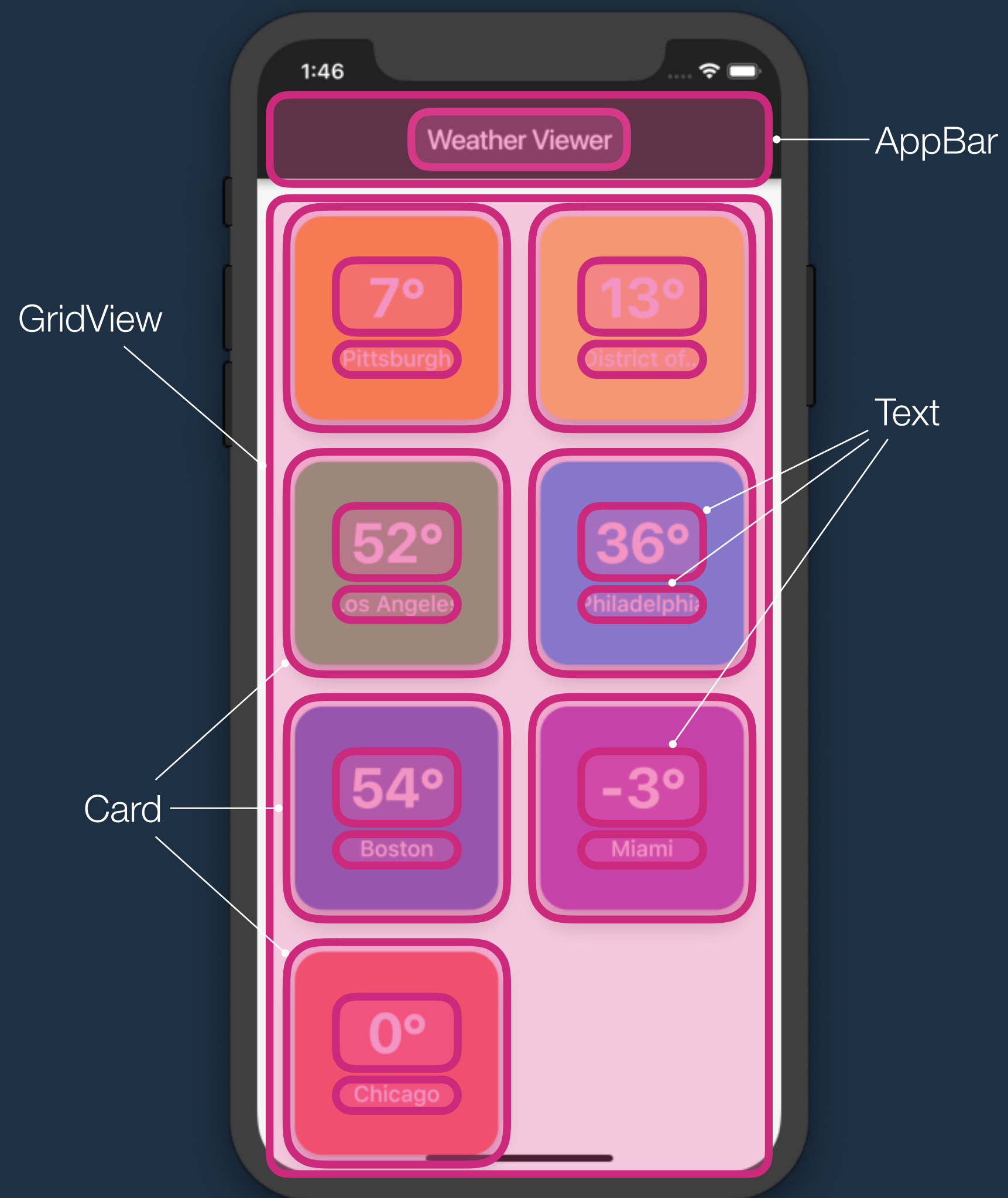


<https://flutter.io/docs/get-started/install>

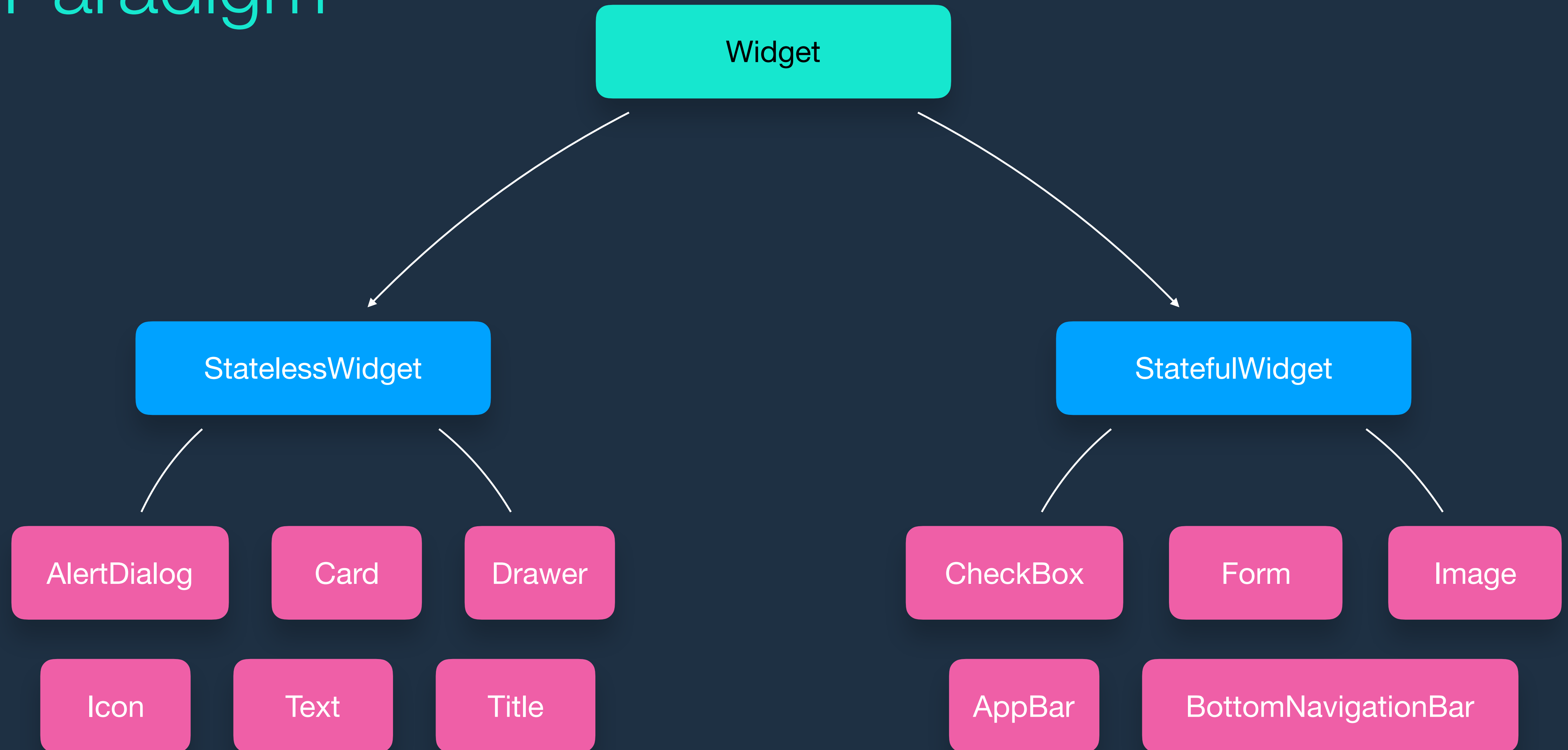
<https://github.com/akshathjain/FlutterWorkshop>

Paradigm

Paradigm



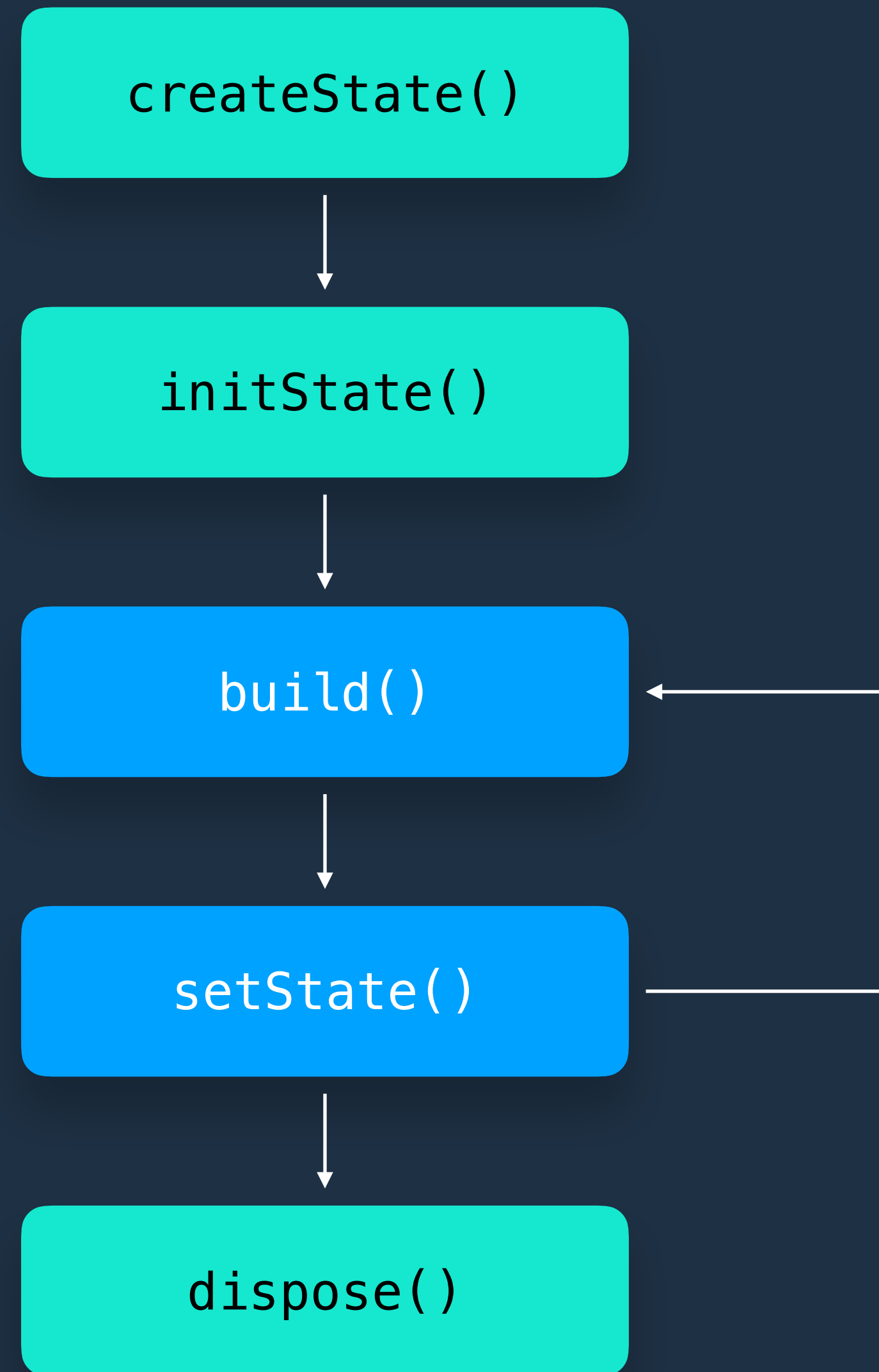
Paradigm



StatefulWidget

Lifecycle

Lifecycle



Hello World

Setup

→ flutter create helloworld

→ cd helloworld

Code

```
helloworld
├── android/
├── ios/
├── lib/
│   └── main.dart
├── test/
└── pubspec.yaml
```

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

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: "Hello World",
      home: HomePage(),
    );
  }
}

class HomePage extends StatefulWidget {
  HomePage({Key key}) : super(key: key);

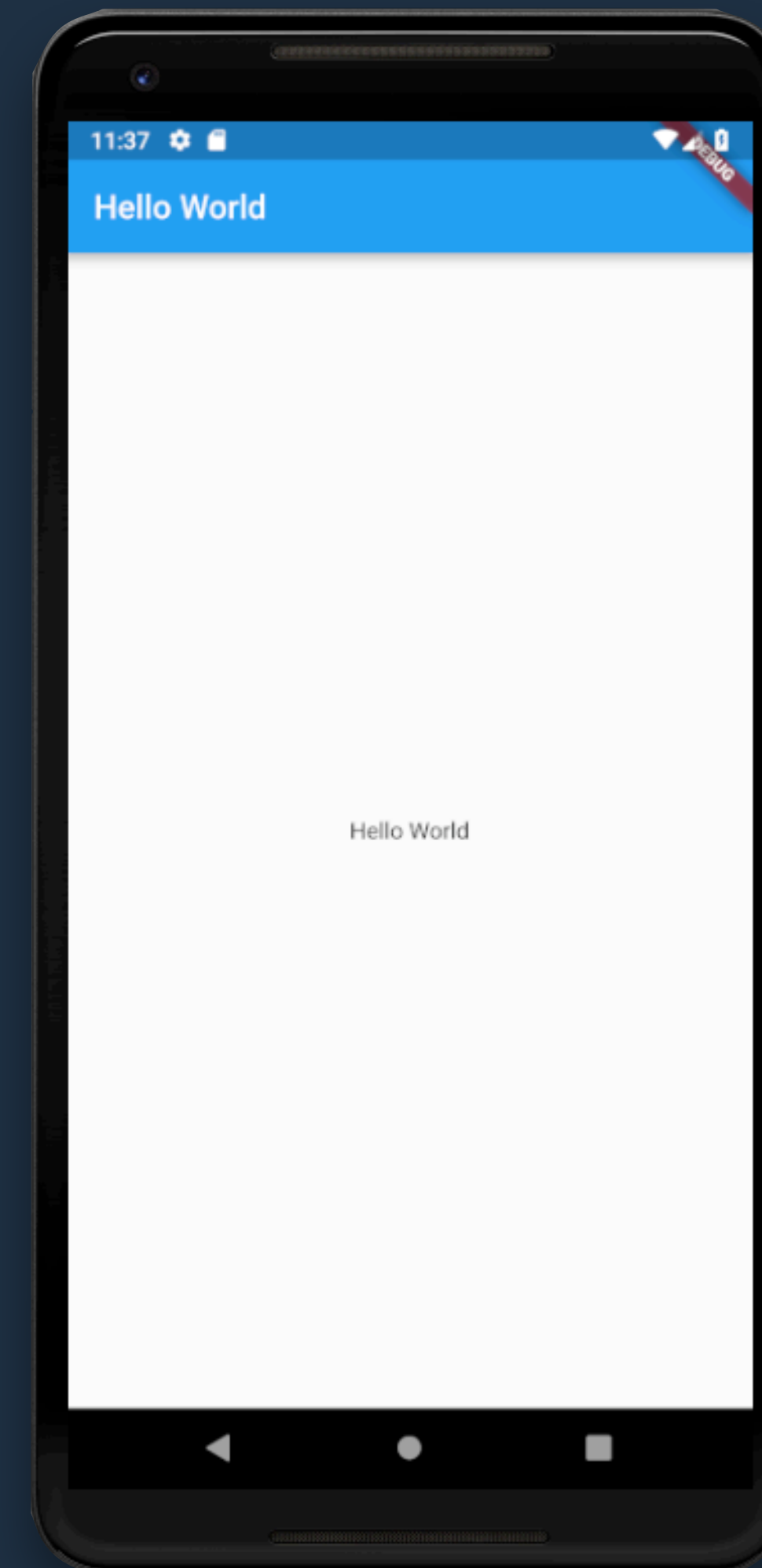
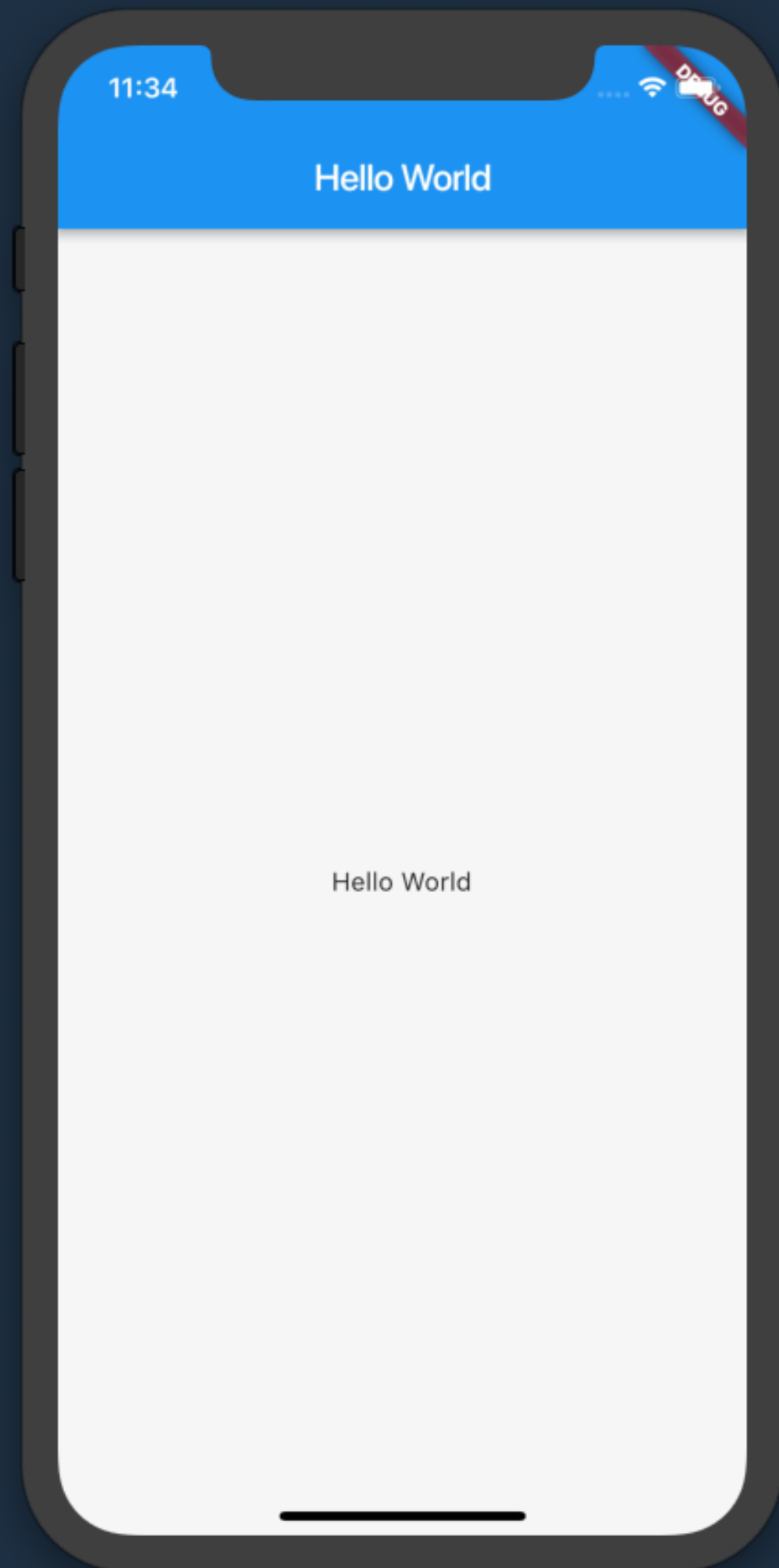
  @override
  _HomePageState createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  @override
  Widget build(BuildContext context){
    return Scaffold(
      appBar: AppBar(
        title: Text("Flutter Demo"),
      ),
      body: Center(
        child: Text("Hello World"),
      ),
    );
  }
}
```

Execution

- `flutter emulators --launch <emulator_id>`
- `flutter run`

Emulation



Live Demo

Live Demo

- `git clone https://github.com/akshathjain/FlutterWorkshop`
- `cd FlutterWorkshop/weatherviewer_boilerplate`

Fin.