Introdução ao Flutter

Orlando Eduardo Pereira
BCET-Computação-UFRB
Recode Jr. - DAF
Mobile Developer

Quem sou?

- Estudo BCET/Eng. De Computação
- Diretor Administrativo Financeiro - Recode Jr
- Mobile Developer
- 1 ano com java
- 6 meses com Flutter
- Instagram: @smaaalll
- GithHub: OrlandoEduardo101

O que é Flutter??

Flutter é um SDK de código aberto criado pelo Google para o desenvolvimento de aplicativos para Android, iOS, Desktop ou Web, além de ser o principal método de criação de aplicativos para o Google Fuchsia.

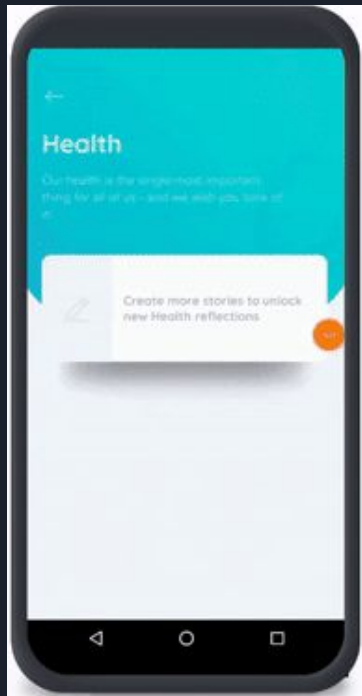
O Flutter é:

- Um framework reativo moderno,

- Um mecanismo de renderização em 2D rápido,

- Ferramentas para desenvolvimento

- Widgets prontos, que constituem os componentes da IU do aplicativo.



Qual a linguagem?

- Originalmente se chamava Dash
- Linguagem de script para web
- Lançada em 2011
- Substituir o JavaScript
- Pode ser executada em VM ou compilada para JS
- Dart 1.0 lançado em Nov. de 2013, primeira vesão estável
- Dart 2.0 lançado em Ago. 2018, Reboot
- Otimizado para Web e Móveis

Porque usar Flutter?

- Desenvolvimento rápido
- Interfaces super bonitas
- Performance nativa
- Criado e mantido pelo Google e pela comunidade
- Comunidade ativa e crescendo
- Mais de 170 widgets (componentes) prontos para serem utilizados
- Principais plugins para acesso à recursos nativos do celular (bateria, câmera, conectividade, webview, etc) também são mantidos pelo Google
- Flutter desenha todos os pixels na tela, tornando o aplicativo altamente customizável
- Alta performance: Aplicativos rodam em 60 frames por segundo (ou em até 120, caso o aparelho suporte)
- Alta produtividade. Alterações no código refletidas no celular ou emulador em até 0,5s. Caso precise reiniciar por completo o app, isto é feito em menos de 2 segundos

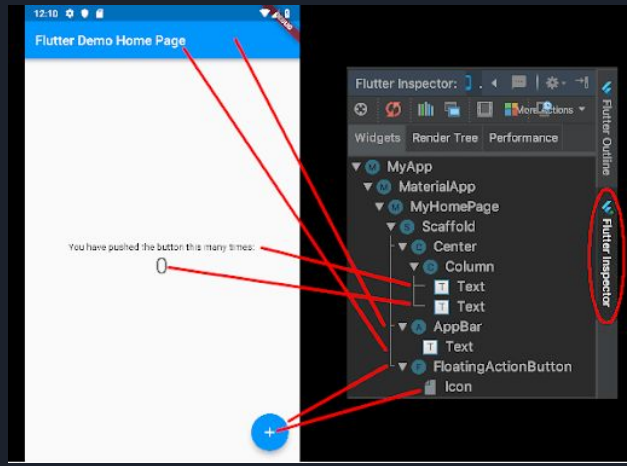
Algumas considerações sobre Dart

- Tipada, porém isto é opcional.
- Morderna (generics<T>, funções, interfaces e mixins).
- High Order Functions (forEach(), map(), etc).
- Utiliza sintáxe C-style (como C#, Javascript).
- Pode ser compilada e interpretada.
- testes : https://dartpad.dev

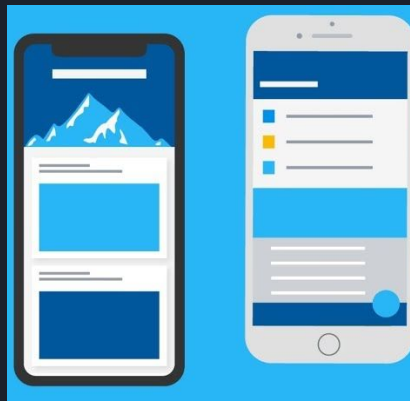
```dart
var A = "Fala Zezé,";
String B = "Bom dia cara.";

void main() {
    print(A + B);
}
```

```
Console

  Fala Zezé,Bom dia cara.
```

“No Flutter, tudo é um widget”

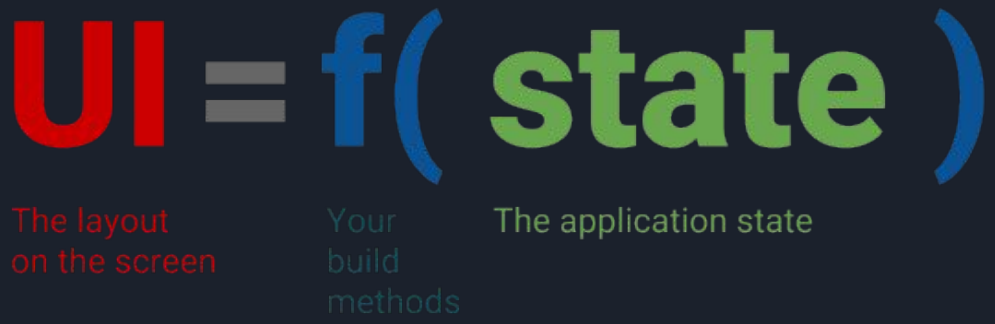


- Widget: É um componente do seu aplicativo, seja um botão, um ícone, um texto, uma imagem, etc. Ao final, vários widgets irão compor o nosso aplicativo.
- Widget tree: É a estrutura que representa como nossos widgets estão organizados.
- pub.dev: Gerenciador de pacotes da linguagem Dart.
- Package: Um módulo/pacote, puramente com código Dart.
- Plugin: Contém código específico da plataforma (java/kotlin, swift/obj-c), geralmente faz acesso à algum recurso nativo, como bateria, câmera.
- Material package: Conjuntos de widgets que seguem as definições do Material Design.
- Cupertino package: Conjunto de widgets que seguem as definições de interface do iOS.



Declarative UI

Nossa interface, reflete o estado da nossa aplicação. Sempre que o estado da aplicação muda, os widgets são reconstruídos para atender aquele novo estado. Seja um botão ou ícone que muda de cor após ser pressionado ou até mesmo toda uma tela após o usuário logar no aplicativo.

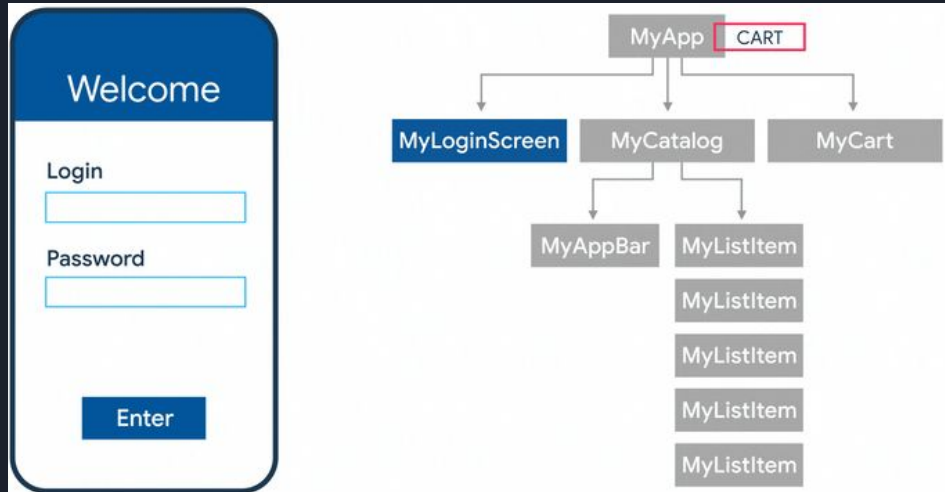


Composição > Herança

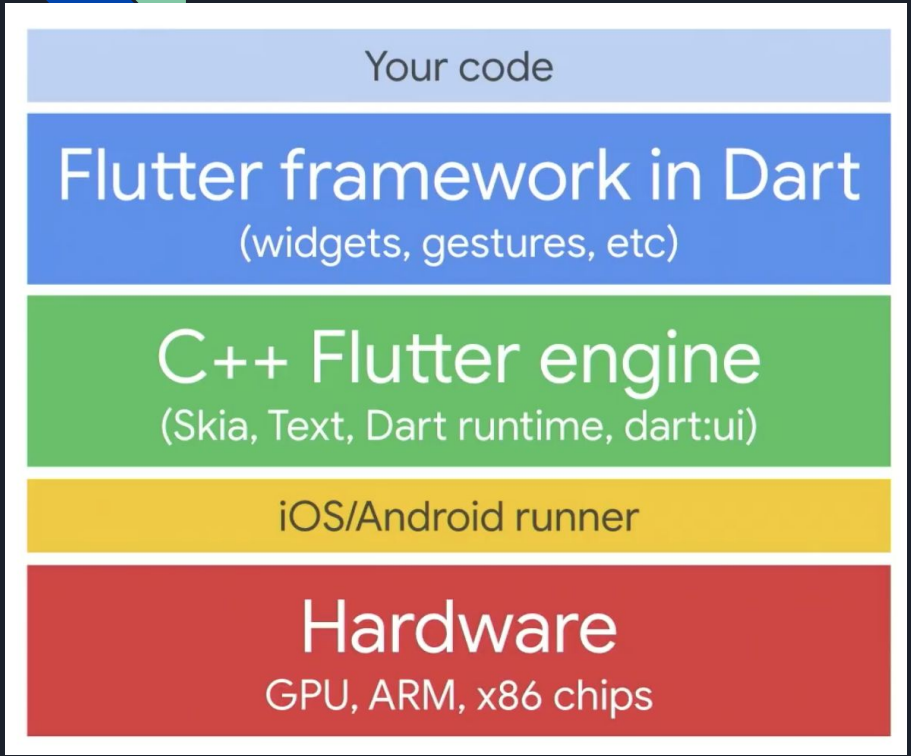
Composição: Widget X possui o widget Y.

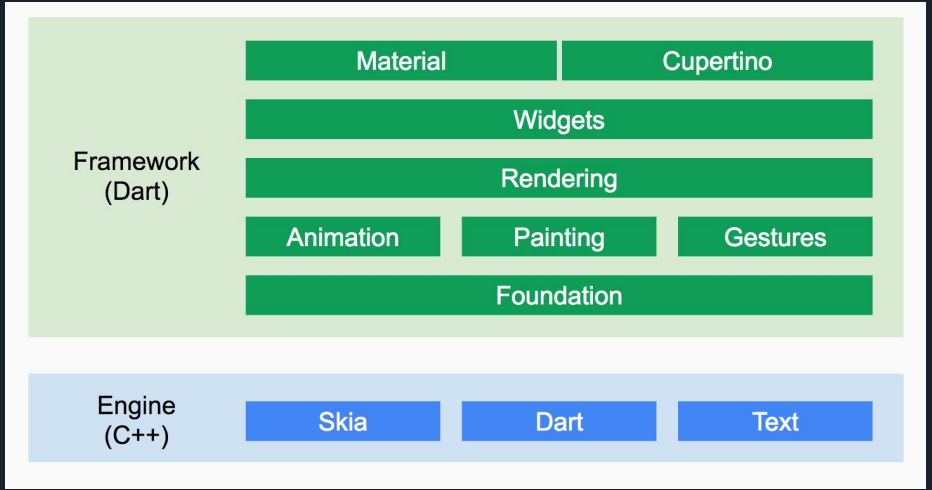
Herança: Widget X é um widget Y (já que X é filho de Y).

Lembrando que widgets são pequenos "blocos", fica simples de entender o motivo de Flutter utilizar esse conceito. Conforme formos montando vários blocos, nosso aplicativo vai tomando forma. Esta montagem dos blocos é a composição.



Arquitetura



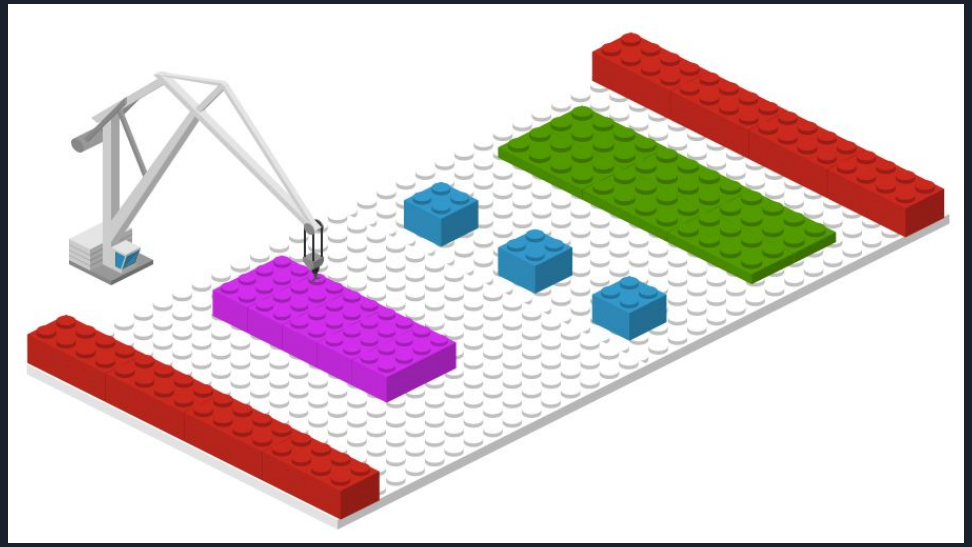


- Your code: Dart, Kotlin, Swift, Packages e Plugins
- Flutter: o framework em si.
- Engine: une o seu codigo e o Framework e prepara pro hardware.
  - Usa a biblioteca Skia e o processador de texto do Chrome.
- Runners: delega e controla os binários ao hardware.
- Hardware: Processador, RAM, Chips...

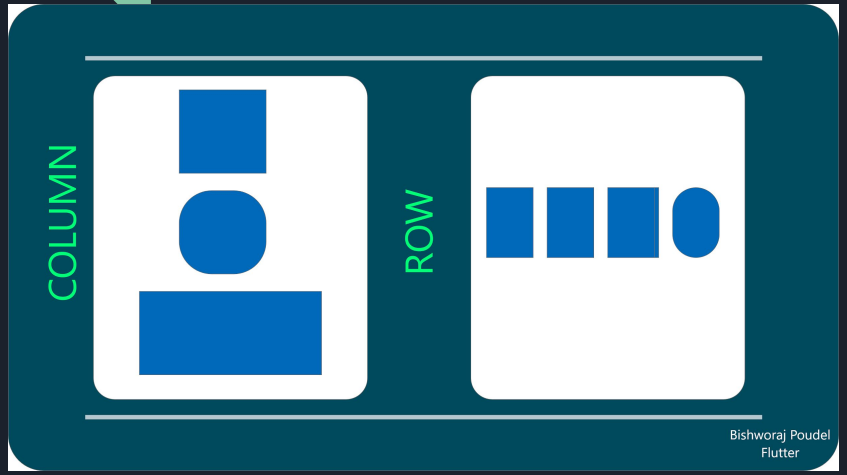
Widgets

"A ideia central é que você construa sua interface com widgets"

- Cada Widget é como uma peça de lego, existem tamanhos diferentes que podem ser acopladas para formar um objeto maior.

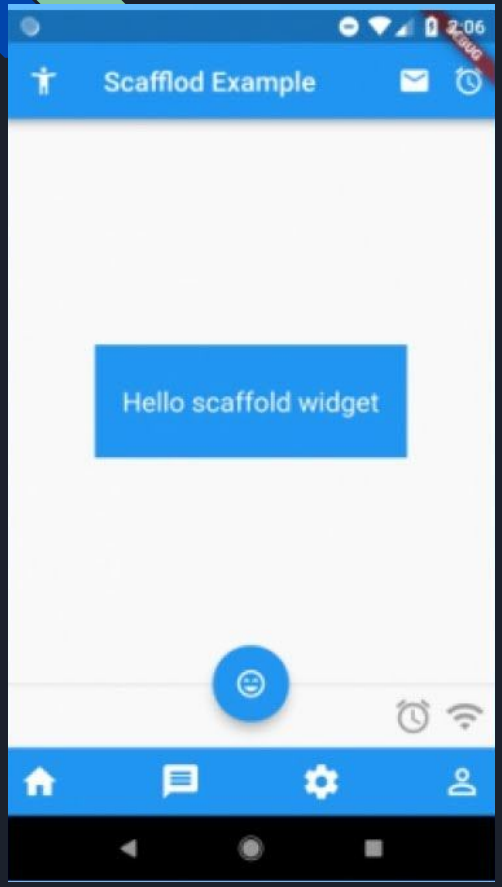


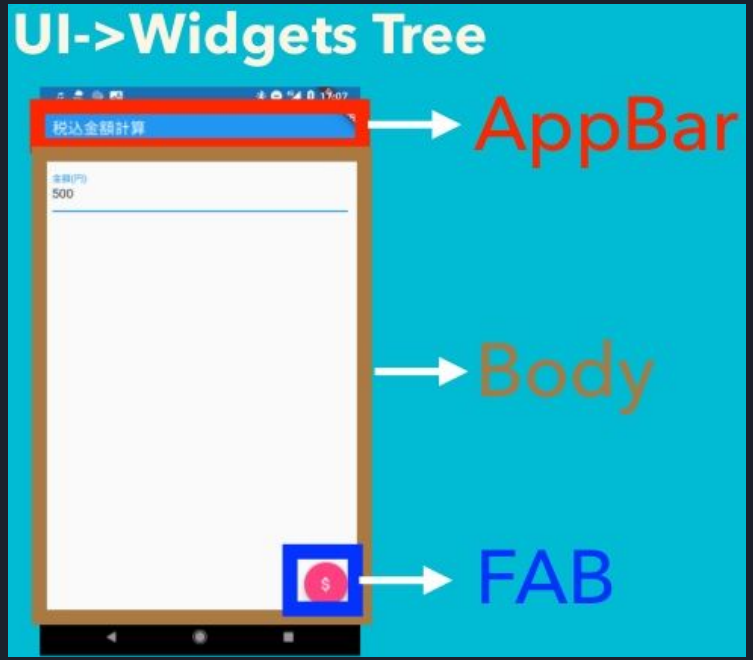
Column & Row



- Widgets de estrutura de layout, neles é possível colocar vários widgets alinhados.

Scaffold (“Andaime”)



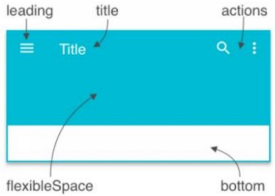


- Estrutura básica padrão para criar um layout rápido
- Material Design
- Possui APIs para construir Snack Bars, Bottom Navigator, Body, Float Action Button...

Scaffold

UI / UX

AppBar



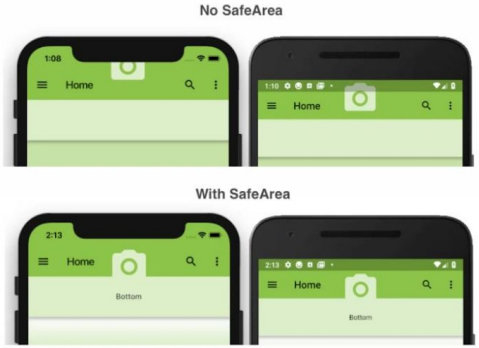
Text

```
onPanUpdate:
DragUpdateDetails(Offset(0.3, 0.0))
```

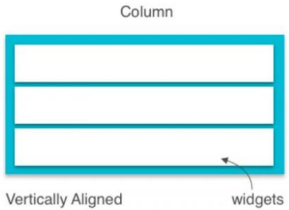
RichText

Flutter World for **Mobile**

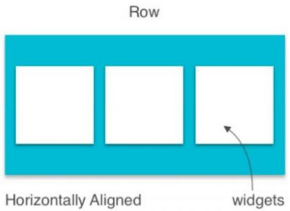
SafeArea



Column



Row



Container

Button

Stateless x Stateful

Stateless: Um widget que não tem seu estado alterado.

- Um texto, é um widget que não tem seu estado alterado.

Stateful: Um widget que tem seu estado alterado.

- Um switch, é um widget que tem seu estado alterado.

Navigator

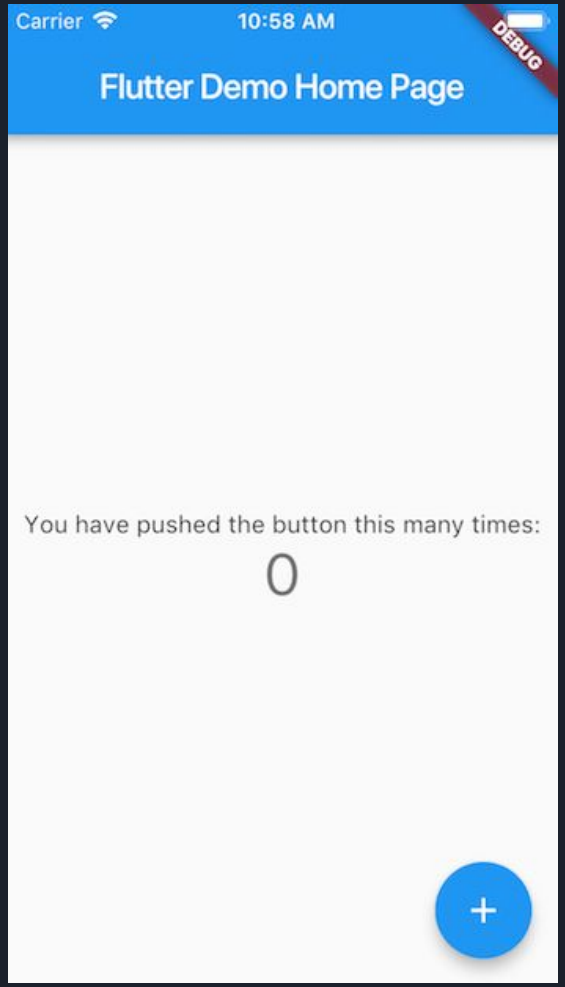
- Push(context, Route):
  Abre a nova tela “em cima” da atual
- Pop(context):
  Fecha a tela “de cima” voltando para a “de trás”

```
// Within the `FirstRoute` widget
onPressed: () {
  Navigator.push(
    context,
    MaterialPageRoute(builder: (context) => SecondRoute()),
  );
}
```

```
// Within the SecondRoute widget
onPressed: () {
  Navigator.pop(context);
}
```

Comandos básicos

- flutter doctor (analisa se as ferramentas estão ok)
- flutter --version (mostra a versão)
- flutter devices (exibe dispositivos disponiveis)
- flutter create meu_app (cria um projeto)
- flutter create --org <com.example> meu_app (cria um projeto)
- flutter run -d <id-device> (executa o app)
- Flutter run -d all (executa em todos devices)
- hot reload e hot restart



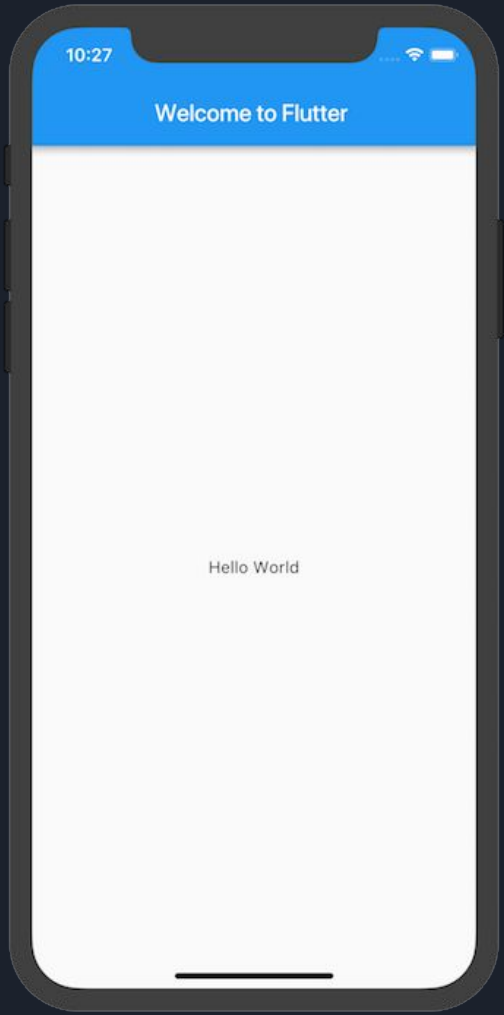
Hello World

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

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Welcome to Flutter',
      home: Scaffold(
        appBar: AppBar(
          title: Text('Welcome to Flutter'),
        ),
        body: Center(
          child: Text('Hello World'),
        ),
      ),
    );
  }
}
```

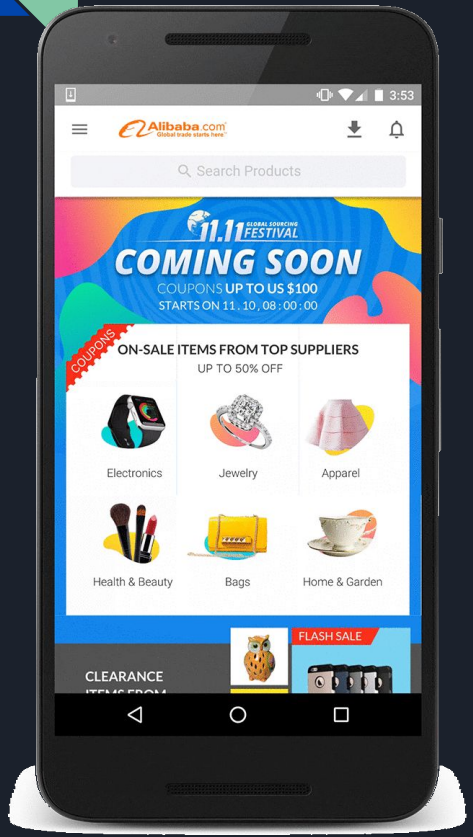
- Documentação

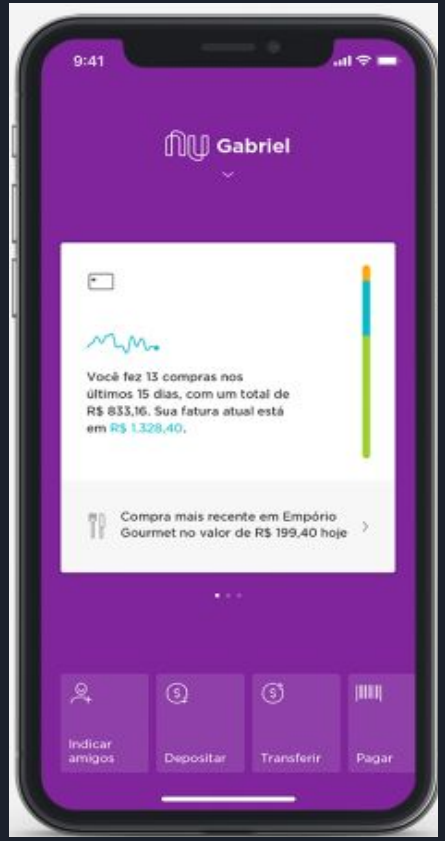
```dart
import 'package:flutter/material.dart';

void main() {
 runApp(
   MaterialApp(
   home: Home(),
 )); // MaterialApp
}

class Home extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Bem vindo ao Flutter"),
      ), // AppBar
      body: Center(
        child: Text("Olá Mundo!"),
      ), // Center
    ); // Scaffold
  }
}
```

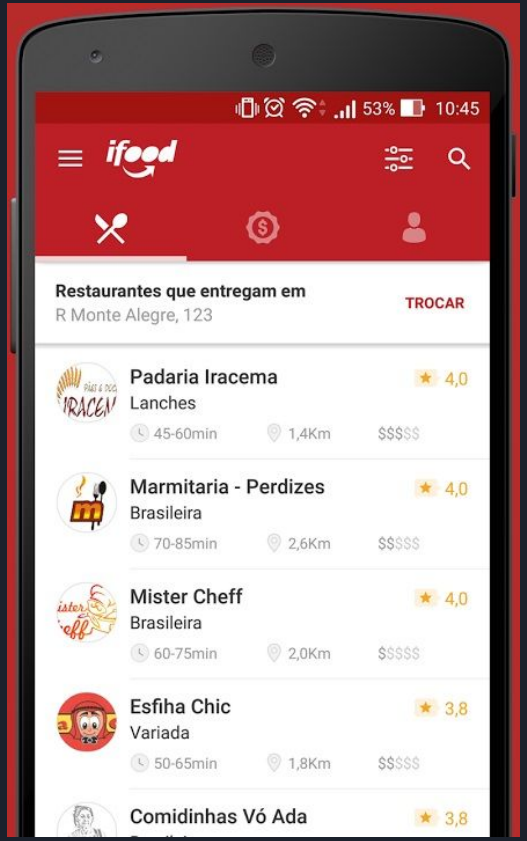
Meu Hello World



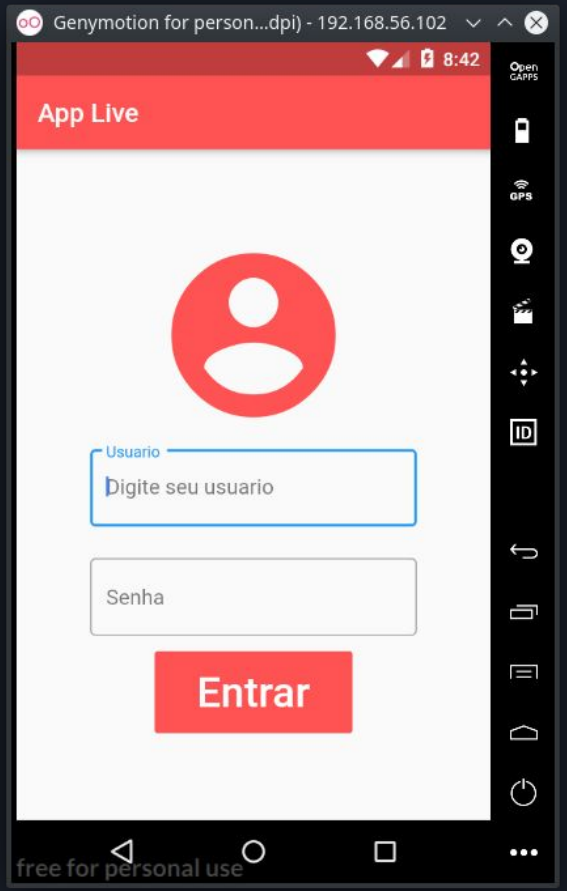
Apps famosos que usam Flutter

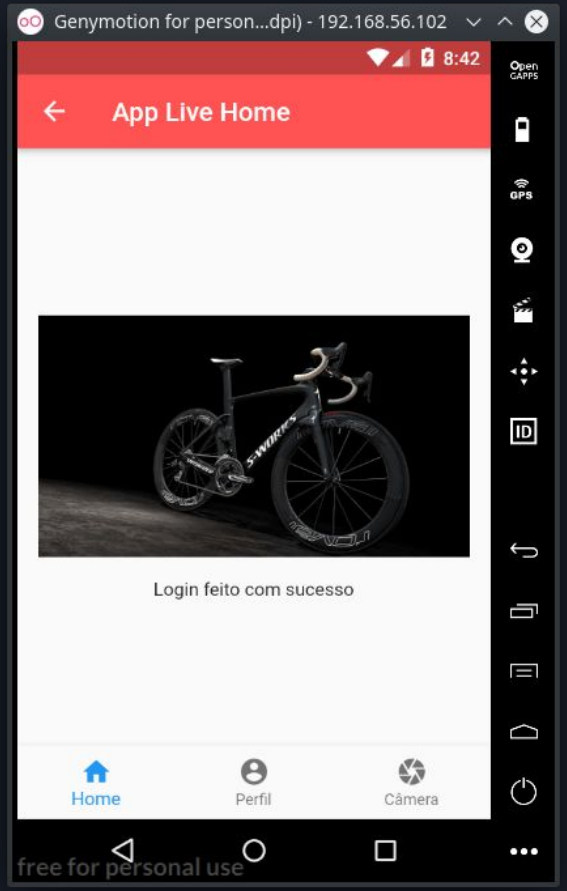






Vamos codar?


Genymotion for person...dpi) - 192.168.56.102
8:42
App Live
Usuario
Digite seu usuario
Senha
Entrar
free for personal use


Genymotion for person...dpi) - 192.168.56.102
8:42
App Live Home
Login feito com sucesso
Home
Perfil
Câmera
free for personal use

Duvidas?

Bibliografia

- https://flutter.dev
- https://dart.dev
- https://dartpad.dev
- https://www.flutterparainiciantes.com.br/o-que-e-flutter
- https://flutterando.com.br/
- https://balta.io/
- https://jamiltondamasceno.com.br/
- https://github.com/OrlandoEduardo101
- https://www.instagram.com/smaaalll/