



Velocidade, praticidade e simplicidade.

O que é?



Flutter é um conjunto de ferramentas para construir aplicações nativas para mobile, web e desktop a partir de um único código.

- Gratuito e de código aberto
- Utiliza Dart - linguagem moderna, orientada a objetos e fácil de aprender
- Fornece widgets para desenhar elementos na tela. São renderizados pela própria engine em alta velocidade.
- Construído com C, C++, Dart, e Skia (motor de renderização 2D)
- Multiplataforma
- Alta performance
- Versão 1.0 foi lançada em dezembro de 2018. Atualmente está na versão 1.9

Quem usa?

GROUPON

nybank

The New York Times



Google Ads

Alibaba Group
阿里巴巴集团

Tencent 腾讯

 **Greentea**

realtor.com[®]

Dart

Dart é uma linguagem de programação lançada em 2011 pela Google com o objetivo de substituir o javascript.

Hoje é descrita como uma linguagem otimizada para UI para construção de apps em qualquer plataforma.

Compila para ARM e x64 para mobile, desktop, backend e JavaScript para web.

Em agosto de 2018 foi lançada a versão 2.0. Até essa apresentação, a versão estável é a 2.5.2.



```
main() async {  
  print('Compute  $\pi$  using the Monte Carlo method.');
```

var output = querySelector("#output");

```
  await for (var estimate in computePi().take(500)) {  
    print('π ≅ $estimate');
```

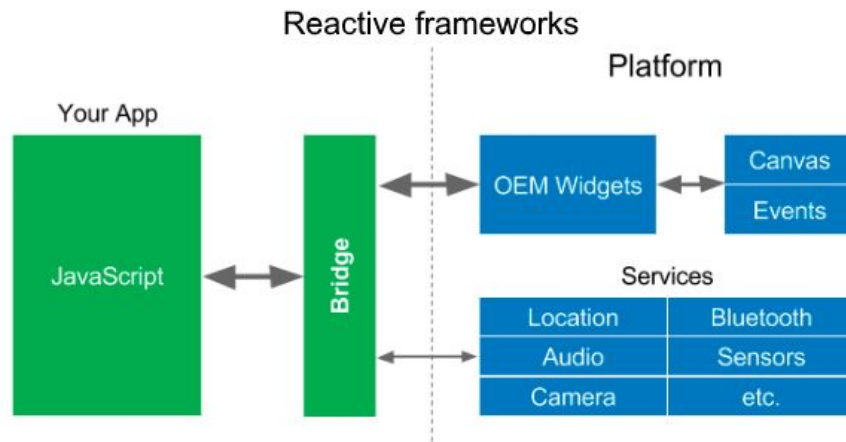
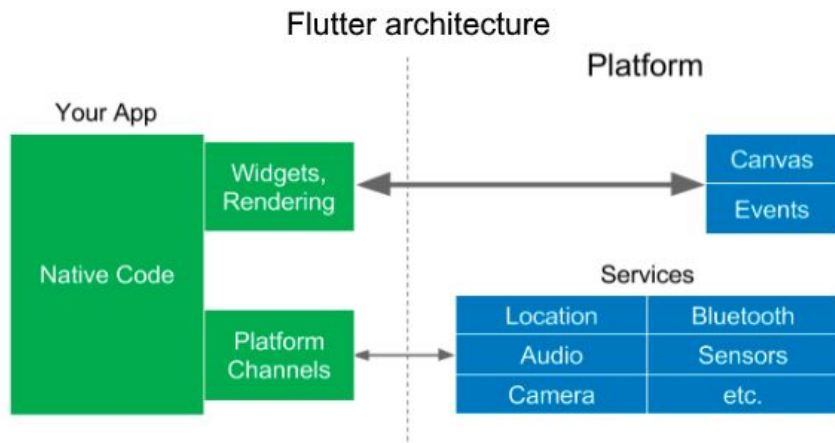
output.text = estimate.toStringAsFixed(5);

```
    await window.animationFrame;  
  }  
}
```

/// Generates a stream of increasingly accurate estimates of π .

```
Stream<double> computePi({int batch: 100000}) async* {  
  var total = 0;  
  var count = 0;  
  while (true) {  
    var points = generateRandom().take(batch);  
    var inside = points.where((p) => p.isInsideUnitCircle);  
    total += batch;  
    count += inside.length;  
    var ratio = count / total;
```

Flutter x React Native x Ionic



AOT, JIT e Hot reload

Ahead-of-Time - com tudo já compilado, a aplicação abre e roda mais rápido. Ótimo para produção mas ruim para desenvolvimento.

Just-in-Time - compila *on the fly* o que torna o desenvolvimento muito mais rápido mas deixa a inicialização do app mais lenta.

Hot reload - atualiza a interface em tempo real quando modificações são feitas no código. Consegue manter o estado da aplicação(não perde os dados de um formulário por exemplo).



It's all Widgets

O Flutter fornece widgets para tudo. Textos, inputs, imagens, containers e tudo mais que compõem a UI é construído a partir de um widget.



```
class _MyHomePageState extends State<MyHomePage> {  
  int _counter = 0;  
  
  void _incrementCounter() {  
    setState(() {  
      _counter++;  
    });  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text(widget.title),  
      ), // AppBar  
      body: Center(  
        child: Column(  
          mainAxisAlignment: MainAxisAlignment.center,  
          children: <Widget>[  
            Text(  
              'You have pushed the button this many times:',  
            ), // Text  
            Text(  
              '$_counter',  
              style: Theme.of(context).textTheme.display1,  
            ), // Text  
          ], // <Widget>[]  
        ), // Column  
      ), // Center  
      floatingActionButton: FloatingActionButton(  
        onPressed: _incrementCounter,  
        tooltip: 'Increment',  
        child: Icon(Icons.add),  
      ), // FloatingActionButton  
    ); // Scaffold  
  }  
}
```


pub.dev

Em <https://pub.dev/flutter> encontramos diversos pacotes para o flutter.

Na página de cada pacote encontramos descrição, instalação, changelog, versões e etc.

Para instalar, basta adicionar o pacote como dependência do projeto no **pubspec.yaml** e rodar um **flutter pub get**.



The Good Parts

- ★ Uma gama de widgets prontos para uso
- ★ Fácil de aprender - seja Dart ou Flutter
- ★ AOT, JIT e Hot reload(stateful)
- ★ Layout é todo construído em Dart. Não precisa de XML
- ★ Otimizado para alcançar facilmente os 60fps
- ★ Portabilidade - Android, iOS, Windows, Mac, Linux e Web
- ★ Internacionalização(i18n), localização(l10n) e acessibilidade(a11y)
- ★ Inclui ferramentas para testes unitários e de integração
- ★ Interoperabilidade - acesso às novas features do SO, comunicação com o nativo e plugins



The Good Parts

- ★ Composição - widgets são criados em pequenas unidades garantindo ao desenvolvedor o máximo de customização
- ★ É possível rodar código Dart em background
- ★ Suporte para Android Studio, IntelliJ IDEA, e VS Code
- ★ Suporte para 2Dimensions e Lottie
- ★ Integração com Bitrise - Compilação, teste e implantação
- ★ Redux, Flux, GraphQL, Firebase, MobX, Bloc, etc...



The Bad Parts

- Falta bibliotecas e pacotes de terceiros
- A Google ou a comunidade são responsáveis por novos widgets
- Dart
- O tamanho final do build é grande. Em uma aplicação super básica chega a ~4,5MB no Android e ~10MB no iOS



Exemplos

Page Transition

Fade Second Page

Left To Right Slide Second Page

Size Slide Second Page

Rotate Slide Second Page

Scale Slide Second Page

Upto Down Second Page

Down To Up Second Page

PushNamed Page

09:41



Journal

Add Food

Vegetables as is

1.0 Small portion

226

41

0

4.5

To which meal?

Lunch2



Search our database by name

Bread

Vegetables as is



How Much?

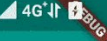
1.0

Small portion



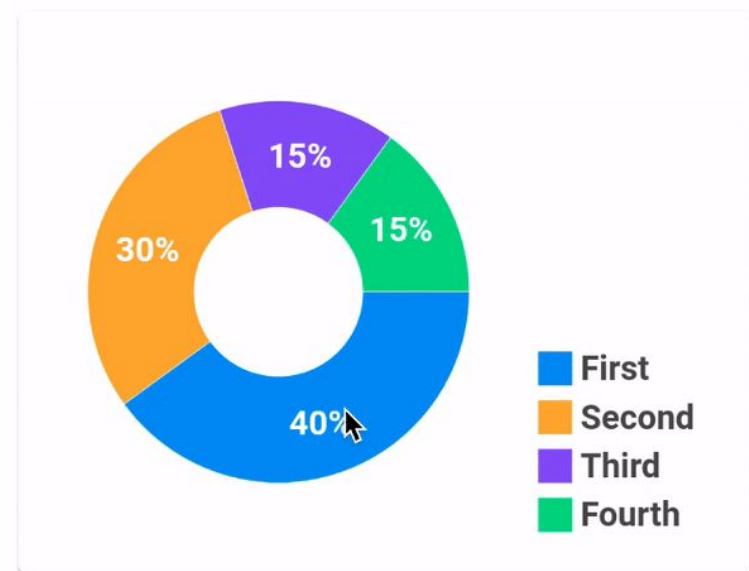
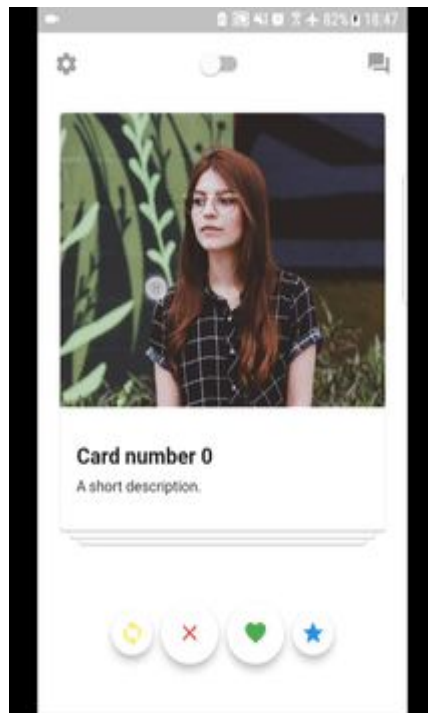
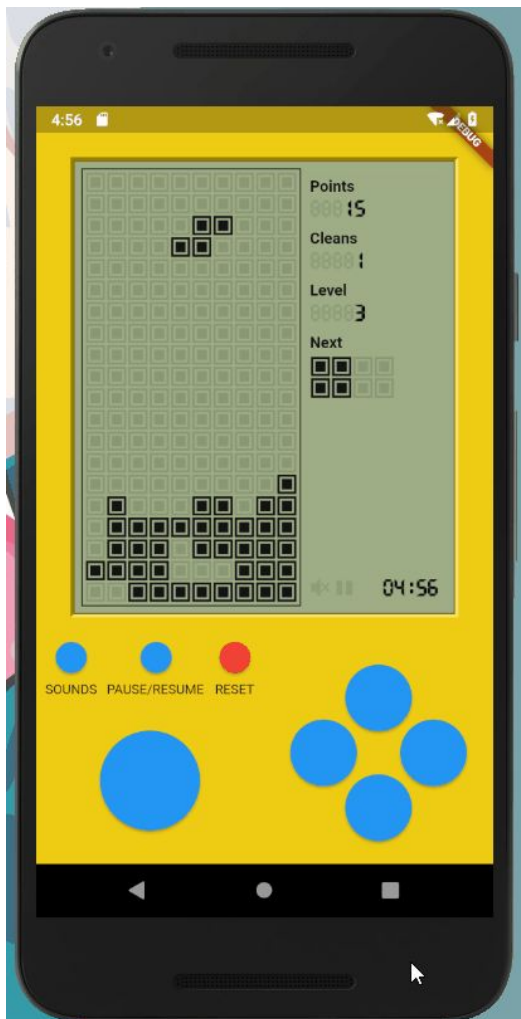
ADD TO JOURNAL

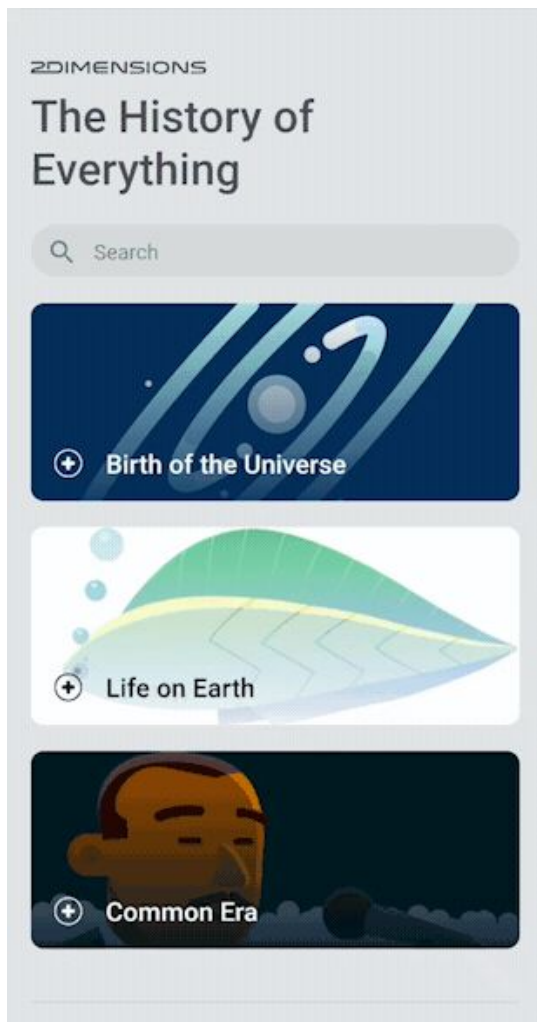
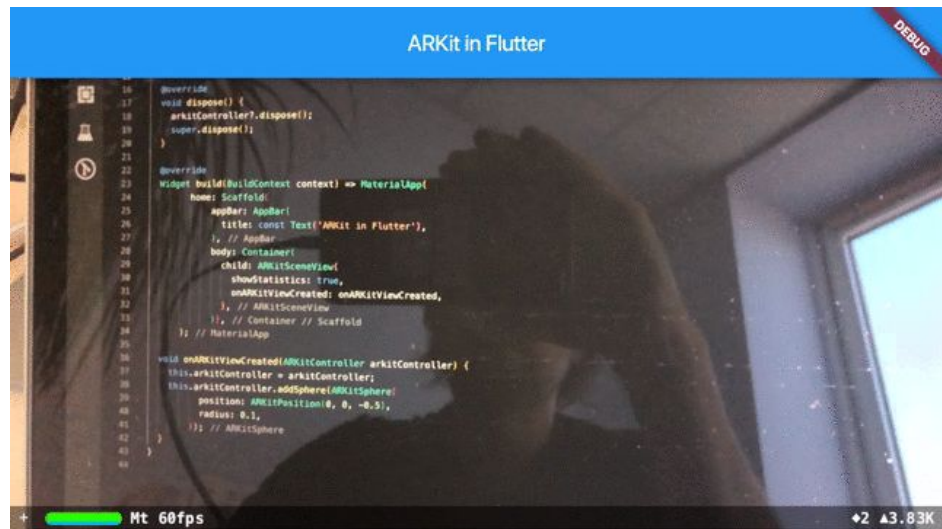
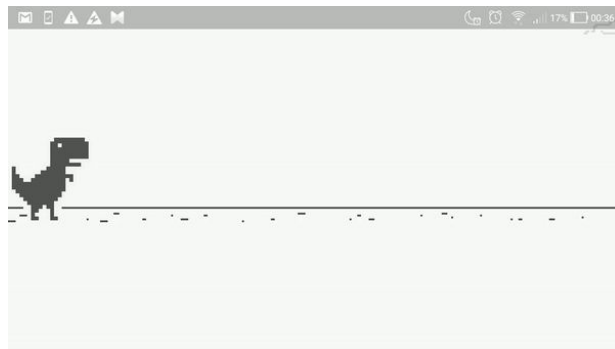
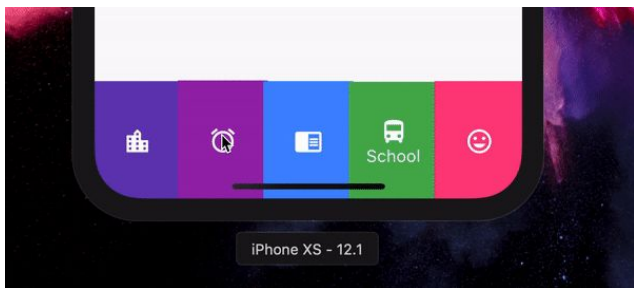
3:22 AM



Before After







Obrigado!

