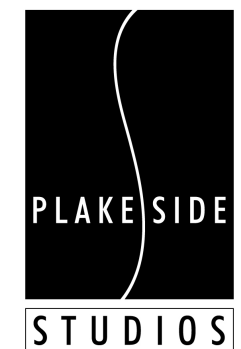


a year with flutter

brandon johnson | @brandon_mn
Google Developer Group – Twin Cities

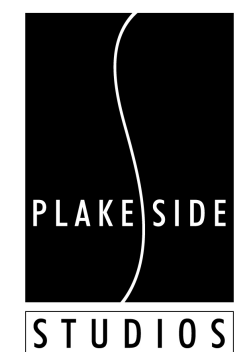
what is flutter?

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



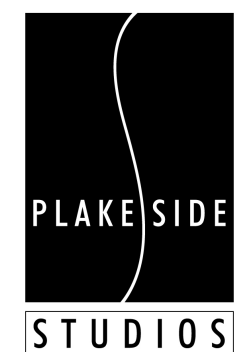
a hybrid mobile app framework

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



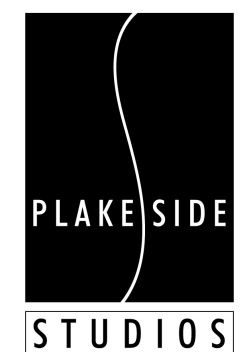
powered by dart

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



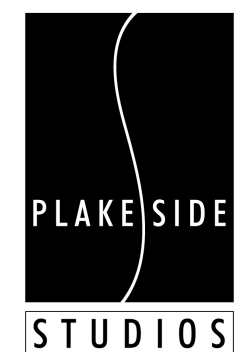
with mostly non- native UI elements

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



and some other neat twists

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



who am I?

Brandon Johnson

@brandon_mn | brandon.mn

I am part of **PlakeSide**, a software shop for the event industry. I (mainly) work on **React Native** and **Flutter** apps, CMSes and backend systems in **Node.js**.

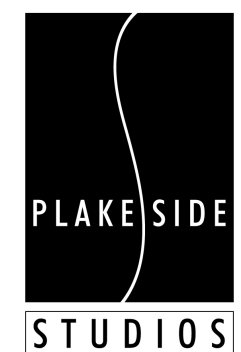
You might also catch me at **JavaScriptMN**, and **ServerlessMN**, which I help run.

Let's get coffee ☕ sometime!



what is flutter?

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020

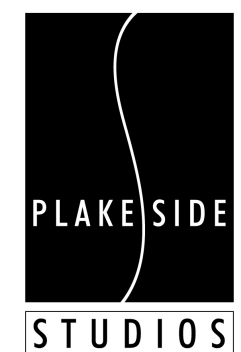


what is flutter?

- a **hybrid mobile app framework** written by **Google**
- **powered by dart** and packaged with a dart VM
- using mostly **non-native, canvas-drawn UI elements**
- **pretty neat!**

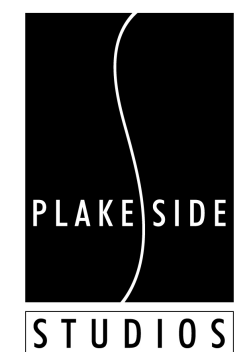
why use flutter?

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



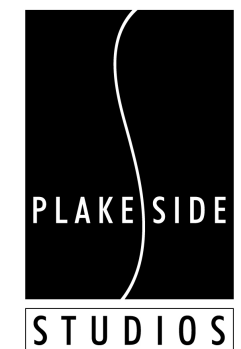
some things i learned

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



widgets

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



Skia Graphics Library

Skia is an open source 2D graphics library which provides common APIs that work across a variety of hardware and software platforms. It serves as the graphics engine for Google Chrome and Chrome OS, Android, Mozilla Firefox and Firefox OS, and many other products.

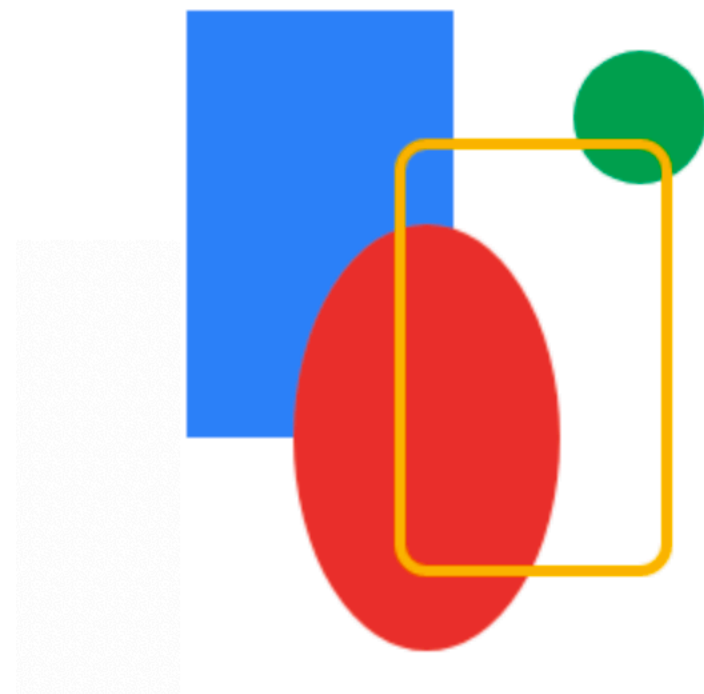
Skia is sponsored and managed by Google, but is available for use by anyone under the BSD Free Software License. While engineering of the core components is done by the Skia development team, we consider contributions from any source.

- Canonical source tree: skia.googlesource.com/skia.
- Issue tracker: bug.skia.org.
- Discussion forum: skia-discuss@googlegroups.com.
- API Reference and Overview: skia.org/user/api.
- Skia Fiddle: fiddle.skia.org.

Showcase

Click on any image below to see the source code that generated the image.

Shapes



Bézier Curves



@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020

Material Components widgets



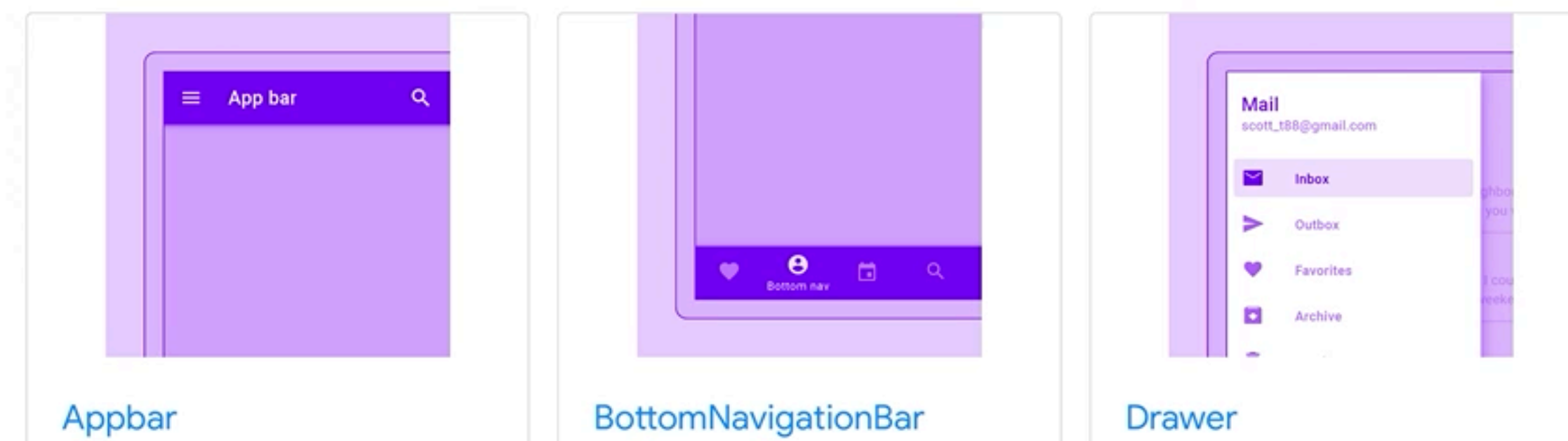
[Docs](#) > [Development](#) > [UI](#) > [Widgets](#) > Material

Visual, behavioral, and motion-rich widgets implementing the [Material Design guidelines](#).

- [App structure and navigation](#)
- [Buttons](#)
- [Input and selections](#)
- [Dialogs, alerts, and panels](#)
- [Information displays](#)
- [Layout](#)

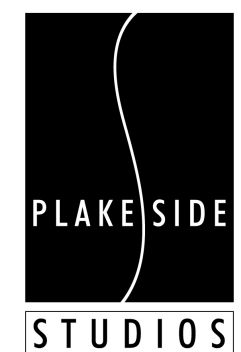
See more widgets in the [widget catalog](#).

App structure and navigation



what are we building?

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



6:03



Flutter Demo Home Page

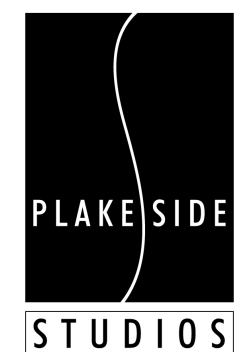
You have pushed the button this many times:

0



stateless widgets

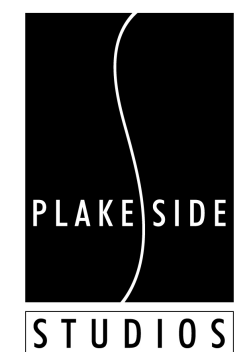
@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



```
class MyApp extends StatelessWidget {  
  // This widget is the root of your application.  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Demo',  
      theme: ThemeData(  
        // This is the theme of your application.  
        primarySwatch: Colors.blue,  
      ), // ThemeData  
      home: MyHomePage(title: 'Flutter Demo Home Page'),  
    ); // MaterialApp  
  }  
}
```

stateful widgets

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



```
class MyHomePage extends StatefulWidget {  
  MyHomePage({Key key, this.title}) : super(key: key);  
  
  final String title;  
  
  @override  
  _MyHomePageState createState() => _MyHomePageState();  
}
```

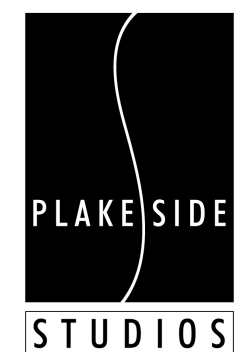
```
class _MyHomePageState extends State<MyHomePage> {
  int _counter = 0;

  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
    );
  }
}
```

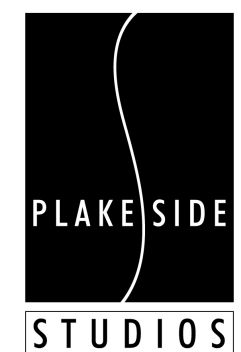
modules

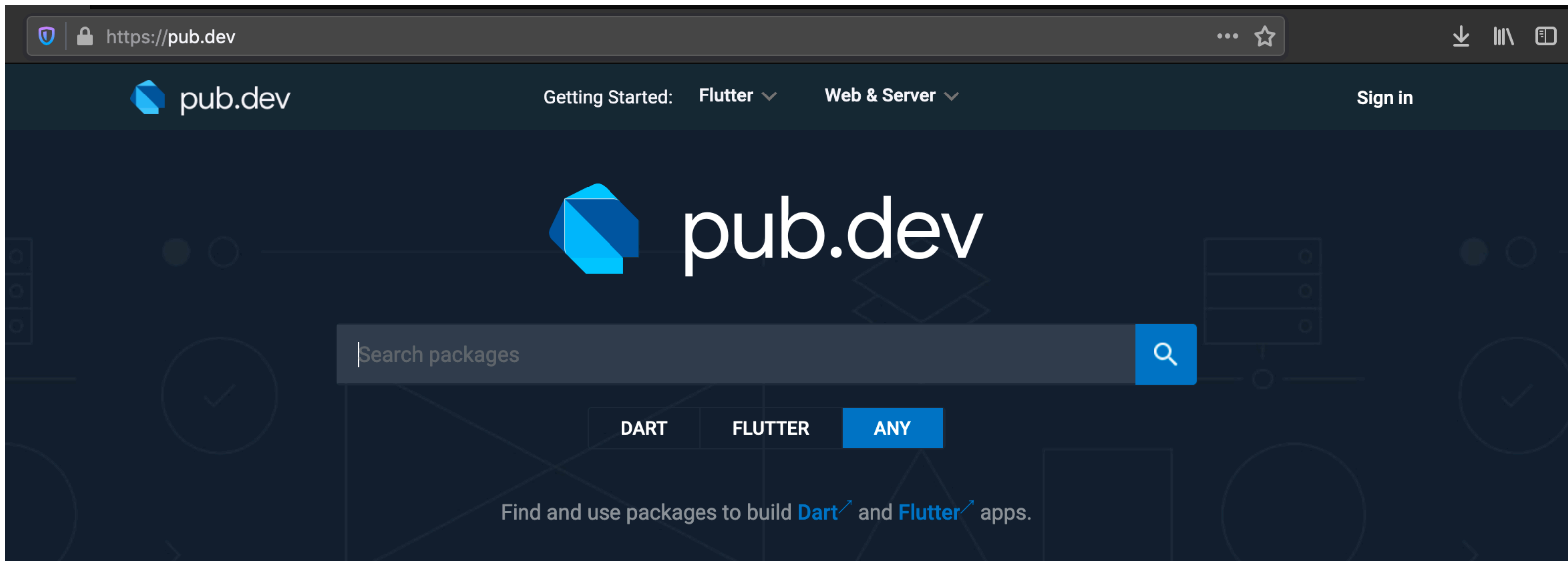
@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



**you get the entire dart
ecosystem****

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020





Flutter Favorites

shared_preferences

flutter.dev

FLUTTER

Flutter plugin for reading and writing simple key-value pairs. Wraps UserDefaults on iOS and SharedPreferences on Android.

mobx_codegen

dart.pixelingene.com

DART

Code generator for MobX that adds support for annotating your code with @observable, @computed, @action and also creating Store classes.

flutter_mobx

dart.pixelingene.com

FLUTTER

Flutter integration for MobX. It provides a set of Observer widgets that automatically rebuild when the tracked observables change.

flutter_redux

brianegan.com

FLUTTER

A set of utility Widgets that Provide and Connect to a Redux Store

redux

fluttercommunity.dev

DART

FLUTTER

Redux is a predictable state container for Dart and Flutter apps

rxdart

fluttercommunity.dev

DART

FLUTTER

RxDart is an implementation of the popular reactiveX api for asynchronous programming, leveraging the native Dart Streams api.

[More Flutter Favorites...](#)

Top packages

after_layout

fluttercommunity.dev

FLUTTER

Execute code after the first layout of your widget has been performed, i.e. after the first

flutter_launcher_icons

fluttercommunity.dev

DART

FLUTTER

A package which simplifies the task of updating your Flutter app's launcher icon

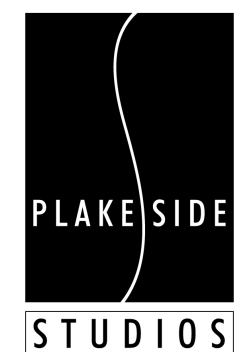
cached_network_image

FLUTTER

Flutter library to load and cache network images. Can also be used with placeholder

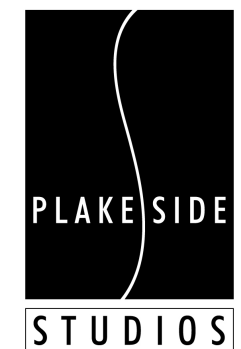
a very active community

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



a double-edged sword

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



webview_flutter
flutter_webview_plugin
???

webview_flutter 0.3.19+5

Published Jan 15, 2020 •  flutter.dev  79 likes

FLUTTER ANDROID IOS

Readme Changelog Example Installing Versions 100

WebView for Flutter (Developers Preview)

pub v0.3.19+5

A Flutter plugin that provides a WebView widget.

On iOS the WebView widget is backed by a [WKWebView](#); On Android the WebView widget is backed by a [WebView](#).

Developers Preview Status

The plugin relies on Flutter's new mechanism for embedding Android and iOS views. As that mechanism is currently in a developers preview, this plugin should also be considered a developers preview.

flutter_webview_plugin 0.3.10+1

Published Dec 28, 2019 •  fluttercommunity.dev  49 likes

FLUTTER ANDROID IOS WEB

Readme Changelog Example Installing Versions 100



Rafal Wachol (@RafalWachol)

Flutter WebView Plugin

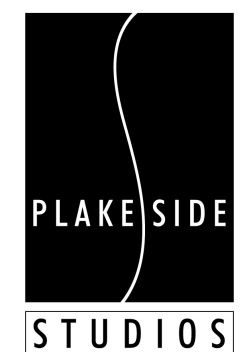
pub v0.3.10+1

Plugin that allows Flutter to communicate with a native WebView.

Warning: The webview is not integrated in the widget tree, it is a native view on top of the flutter view. You won't be able see snackbars, dialogs, or other flutter widgets that would overlap with the region of the screen taken up by the webview.

data flow

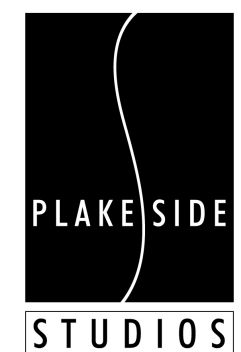
@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



you can use redux
mobx
reactiveX
component state
whatever

Provider pattern

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



provider 4.0.2

Published Jan 18, 2020 ·  dash-overflow.net  454 likes

FLUTTER ANDROID IOS WEB

Readme Changelog Example Installing Versions **100**

build **passing** pub v4.0.2  codecov **98%** chat on gitter



A mixture between dependency injection (DI) and state management, built with widgets for widgets.

It purposefully uses widgets for DI/state management instead of dart-only classes like `Stream`. The reason is, widgets are very simple yet robust and scalable.

By using widgets for state management, `provider` can guarantee:

- maintainability, through a forced uni-directional data-flow
- testability/composability, since it is always possible to mock/override a value
- robustness, as it is harder to forget to handle the update scenario of a model/widget



Publisher

 dash-overflow.net

About

A mixture between dependency injection and state management, built with widgets for widgets.

[Repository \(GitHub\)](#)

[View/report issues](#)

[API reference](#)

License

MIT ([LICENSE](#))

Dependencies

[flutter](#), [nested](#)

More

[Packages that depend on provider](#)

BLoC party 🎉

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



bloc 3.0.0

Published Dec 25, 2019 • [bloclibrary.dev](#)  105 likes

[DART](#) [NATIVE](#) [JS](#) [FLUTTER](#) [ANDROID](#) [IOS](#) [WEB](#)

[Readme](#) [Changelog](#) [Example](#) [Installing](#) [Versions](#) 100

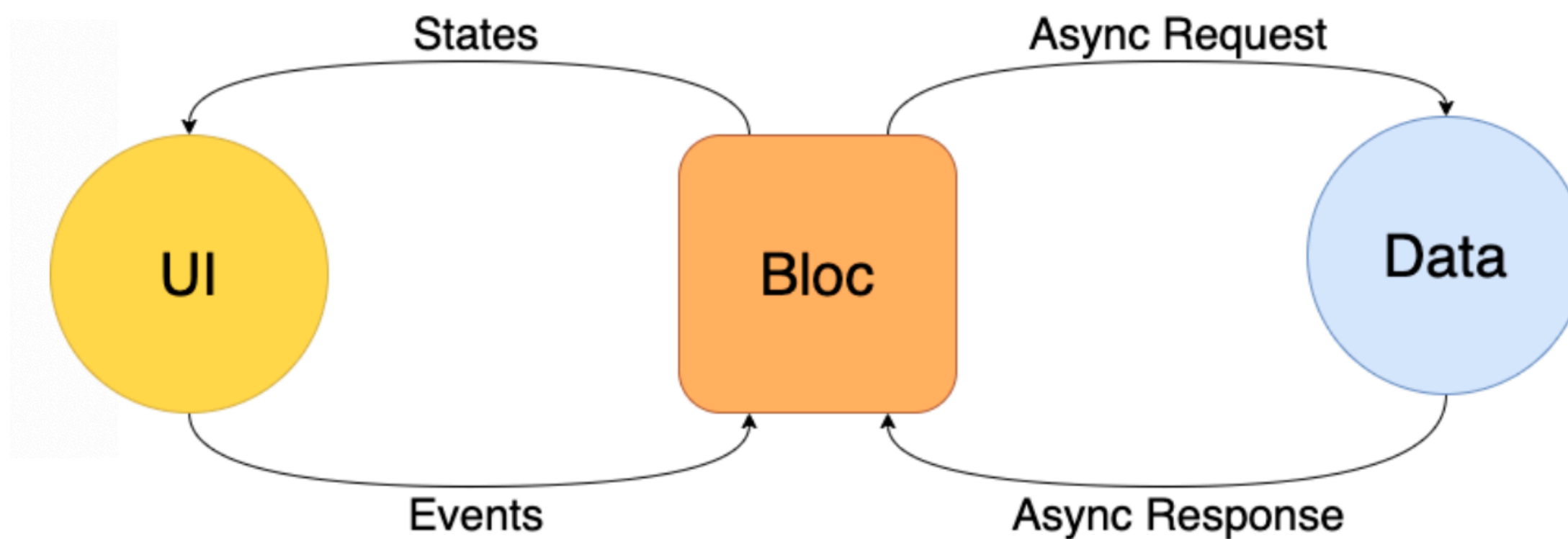
Bloc

pub v3.0.0 build passing codecov 100% style effective dart flutter website awesome flutter
flutter samples stars 3.7k chat 41 online license MIT

A dart package that helps implement the [BLoC pattern](#).

This package is built to work with [flutter_bloc](#) and [angular_bloc](#).

Overview



The goal of this package is to make it easy to implement the [BLoC](#) Design Pattern (Business Logic Component).

Publisher

[bloclibrary.dev](#)

About

A predictable state management library that helps implement the BLoC (Business Logic Component) design pattern.

[Homepage](#)

[Repository \(GitHub\)](#)

[View/report issues](#)

[API reference](#)

License

MIT (LICENSE)

Dependencies

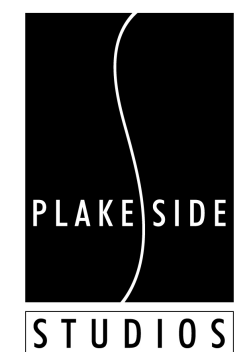
[meta](#), [rxdart](#)

More

[Packages that depend on bloc](#)

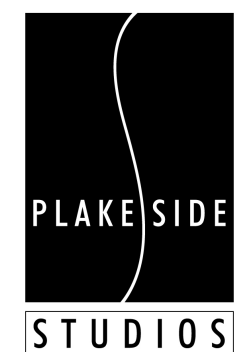
I mostly use the provider pattern

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



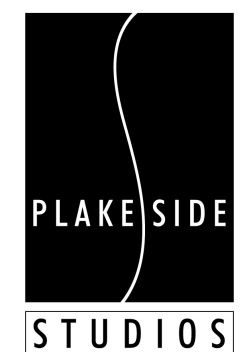
common things you'll need to do


@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



serialize/deserialize JSON

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020





YOU

KNOW

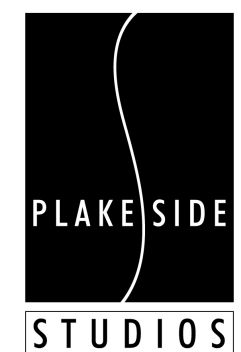
HIS

NAME

{ JASON BOURNE }

verbose, but fine.

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



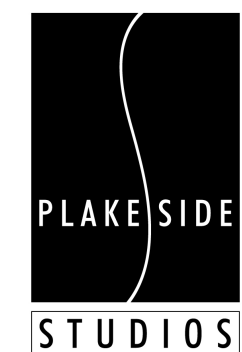
```
class Post {
  final int userId;
  final int id;
  final String title;
  final String body;

  Post({this.userId, this.id, this.title, this.body});

  factory Post.fromJson(Map<String, dynamic> json) {
    return Post(
      userId: json['userId'],
      id: json['id'],
      title: json['title'],
      body: json['body'],
    );
  }
}
```

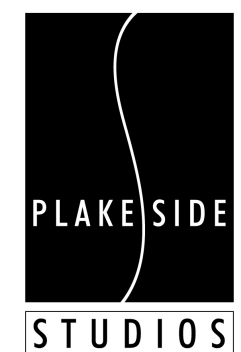

http requests

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



verbose, but fine.

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020

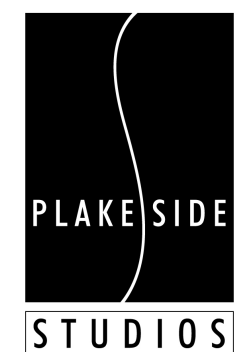


```
Future<Post> fetchPost() async {
  final response =
    await http.get('https://jsonplaceholder.typicode.com/posts/1');

  if (response.statusCode == 200) {
    // If server returns an OK response, parse the JSON.
    return Post.fromJson(json.decode(response.body));
  } else {
    // If that response was not OK, throw an error.
    throw Exception('Failed to load post');
  }
}
```

interaction w/ google services

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020





Documentation

[GUIDES](#)

[GITHUB](#) 

Feedback and Pull Requests are most welcome!

Available FlutterFire plugins

Plugin	Version	Firebase feature	Source code	Web?
cloud_firestore	pub v0.13.0+1	Cloud Firestore	cloud_firestore	
cloud_functions	pub v0.4.1+2	Cloud Functions	cloud_functions	
firebase_admob	pub v0.9.0	Firebase AdMob	firebase_admob	
firebase_analytics	pub v5.0.11	Firebase Analytics	firebase_analytics	
firebase_auth	pub v0.15.4	Firebase Authentication	firebase_auth	
firebase_core	pub v0.4.3+3	Firebase Core	firebase_core	

google_maps_flutter 0.5.22

Published Jan 31, 2020 • flutter.dev 152 likes

FLUTTER ANDROID IOS

Readme

Changelog

Example

Installing

Versions

100

Google Maps for Flutter (Developers Preview)

pub v0.5.22

A Flutter plugin that provides a [Google Maps](#) widget.

Developers Preview Status

The plugin relies on Flutter's new mechanism for embedding Android and iOS views. As that mechanism is currently in a developers preview, this plugin should also be considered a developers preview.

Known issues are tagged with the [platform-views](#) and/or [maps](#) labels.

To use this plugin on iOS you need to opt-in for the embedded views preview by adding a boolean property to the app's `Info.plist` file, with the key `io.flutter.embedded_views_preview` and the value `YES`.

The API exposed by this plugin is not yet stable, and we expect some breaking changes to land soon.

Publisher

flutter.dev

About

A Flutter plugin for integrating Google Maps in iOS and Android applications.

[Homepage](#)

[Repository \(GitHub\)](#)

[View/report issues](#)

[API reference](#)

License

BSD ([LICENSE](#))

Dependencies

[flutter](#),

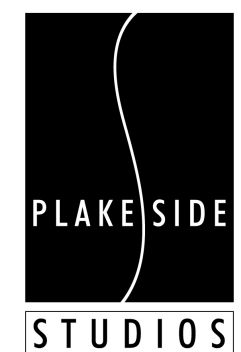
[flutter_plugin_android_lifecycle](#)

More

[Packages that depend on google_maps_flutter](#)

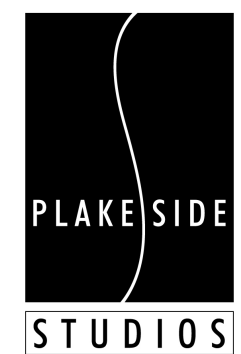
interaction w/ non-google services

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



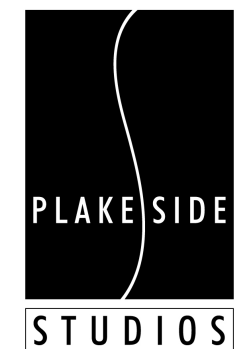


@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



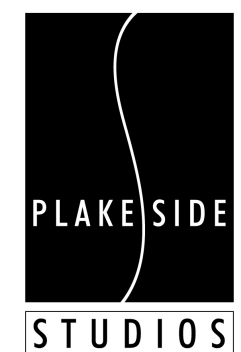
developer tools

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



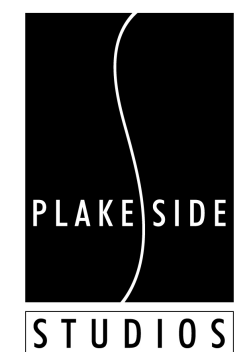
testing frameworks

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



built-in testing library

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



4. Write a test for our class

Inside the `counter_test.dart` file, write the first unit test. Tests are defined using the top-level `test` function, and you can check if the results are correct by using the top-level `expect` function. Both of these functions come from the `test` package.

```
// Import the test package and Counter class
import 'package:test/test.dart';
import 'package:counter_app/counter.dart';

void main() {
  test('Counter value should be incremented', () {
    final counter = Counter();

    counter.increment();

    expect(counter.value, 1);
  });
}
```




4. Build the widget using the `WidgetTester`

Next, build `MyWidget` inside the test environment by using the `pumpWidget()` method provided by `WidgetTester`. The `pumpWidget` method builds and renders the provided widget.

Create a `MyWidget` instance that displays “T” as the title and “M” as the message.

```
void main() {  
  testWidgets('MyWidget has a title and message', (WidgetTester tester) async {  
    // Create the widget by telling the tester to build it.  
    await tester.pumpWidget(MyWidget(title: 'T', message: 'M'));  
  });  
}
```



Note

After the initial call to `pumpWidget()`, the `WidgetTester` provides additional ways to rebuild the same widget. This is useful if you're working with a `StatefulWidget` or animations.

2. Add the `flutter_driver` dependency

Next, use the `flutter_driver` package to write integration tests. Add the `flutter_driver` dependency to the `dev_dependencies` section of the app's `pubspec.yaml` file.

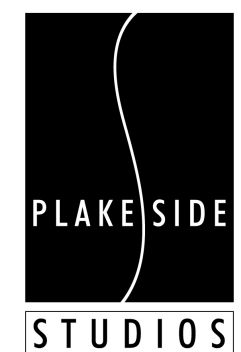
Also add the `test` dependency in order to use actual test functions and assertions.

```
dev_dependencies:  
  flutter_driver:  
    sdk: flutter  
  test: any
```



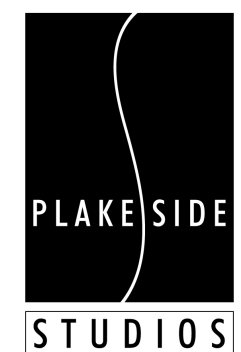
**the best hot reloading
there is!**

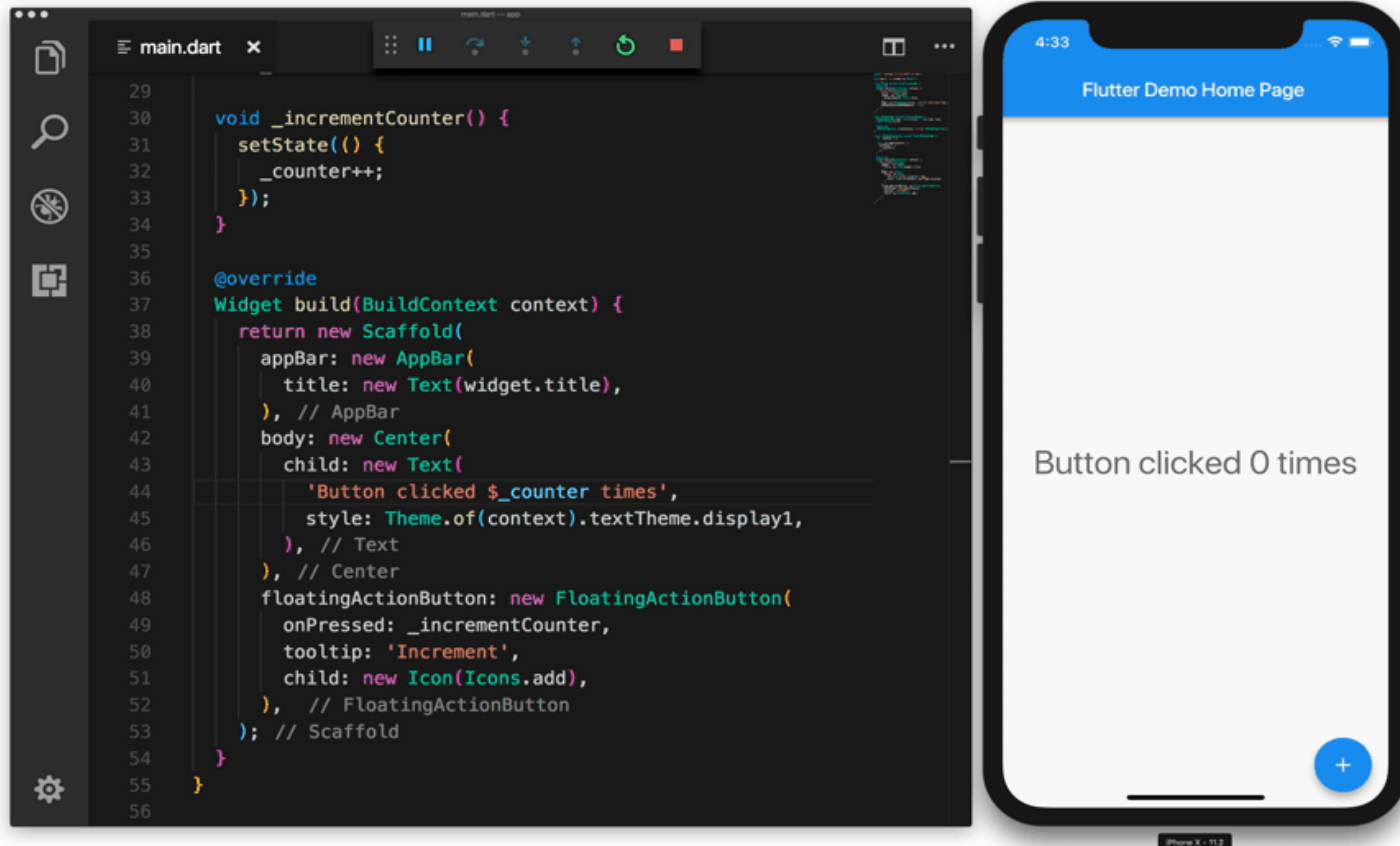
@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



**editor support is
pretty neat too!**

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020

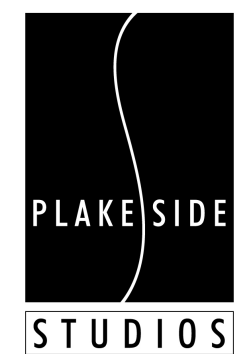




@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020

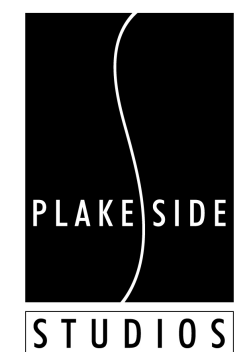
tl;dr

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



why use flutter?

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020



thanks!

@brandon_mn
Google Developer Group Twin Cities
04 Feb 2020

