

# Recommended packages

1. State Management
2. Navigation & Router
3. Forms
4. UI Layout & functions
5. Data & Network
6. Firebase and Google services
7. Communication with Native and Plugins
8. Packages for development

## 1. State Management

- 1.1. Mobx
- 1.2. Riverpod

## 1.1. Mobx

MobX is a library for reactively managing the state of your applications. Use the power of observables, actions, and reactions to supercharge your Dart and Flutter apps.

## 1.1. Mox

### Core Concepts

- Observables - represent the reactive-state of your application.
  - `@observable`
  - `@readonly`
  - `@computed`
- Actions - how you mutate the observables.
- Reactions - They are the observers of the reactive-system and get notified whenever an observable they track is changed.

## 1.1. Mobx

### Reference:

- [Flutter MobX Tutorial – Transparent & Reactive State Management?](#) - Dec 27, 2019
- [Home page](#)

## 1.2. Riverpod

### Reference:

- [Flutter State Management with Riverpod: The Essential Guide](#) - Feb 18, 2021
- [Riverpod Simplified | Easy State Management](#) - Feb 5, 2021
- [Flutter Riverpod Tutorial - The Better Provider](#) - Nov 28, 2020
- [Riverpod.dev](#)

## 2. Navigation & Router

- 2.1. Fluro
- 2.1. VRouter

## 2.1. fluro

Fluro is a null-safe Flutter routing library that adds flexible routing options like wildcards, named parameters and clear route definitions.

### Features:

- Simple route navigation
- Function handlers (map to a function instead of a route)
- Wildcard parameter matching
- Querystring parameter parsing
- Common transitions built-in
- Simple custom transition creation
- Follows stable Flutter channel
- Null-safety

## 2.2. VRouter

A Flutter package that makes navigation and routing easy, inspired by Vue router.

Learn more at [vrouter.dev](https://vrouter.dev)

Here are a few things that this package will make easy:

- Automated web url handling
- Nesting routes
- Transition
- Advanced url naming
- Reacting to route changing
- Customizable pop events
- And much more...

## 3. Forms

- 3.1. flutter\_form\_builder

### 3.1. [flutter\\_form\\_builder](#)

This package helps in creation of data collection forms in Flutter by removing the boilerplate needed to build a form, validate fields, react to changes, and collect final user input.

Video reference:

- [How to create Forms in Flutter \(Flutter Form Builder Tutorial\)](#) - Feb 12, 2021

### 3.1. flutter\_form\_builder

#### Functions:

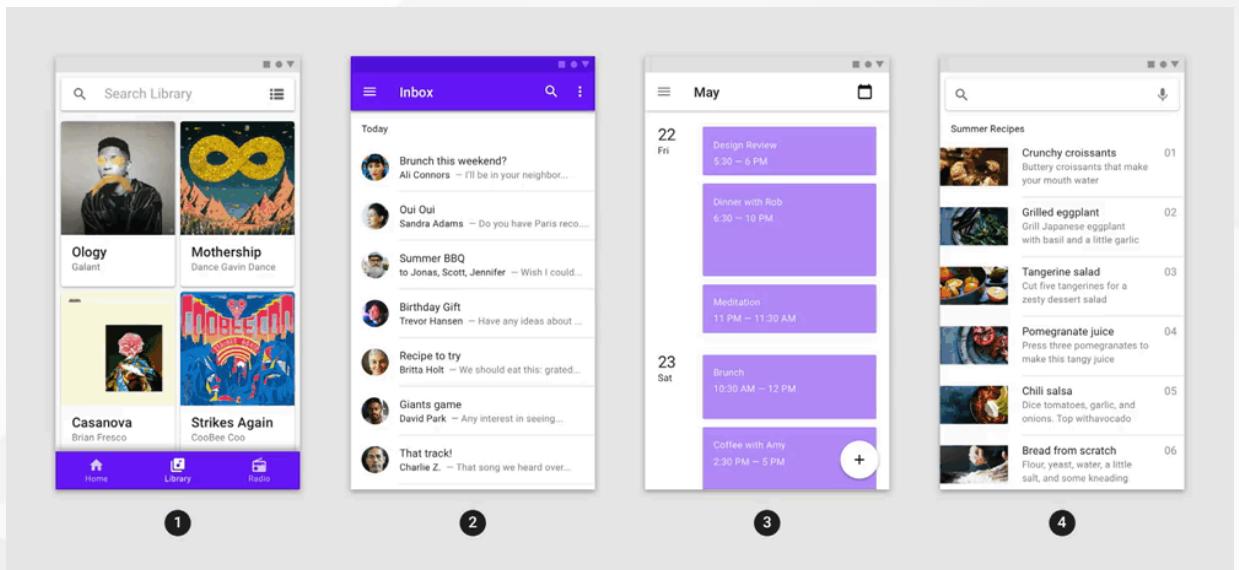
- Form basics supports
  - Additional input fields (as additional packages)
  - Building your own custom field
  - Programmatically changing field value
  - Validation
    - Built-in Validators
    - Using multiple validators
    - Programmatically inducing an error
    - Conditional validation
- You can also validate a field based on the value of another field

## 4. UI Layout & functions

- 4.1. animations
- 4.2. after\_layout
- 4.3. flutter.blurhash
- 4.4. storybook\_flutter
- 4.4. flutter\_typeahead
- ...

### 4.1. animations

Fancy pre-built  
animations that can  
easily be integrated  
into any Flutter  
application.

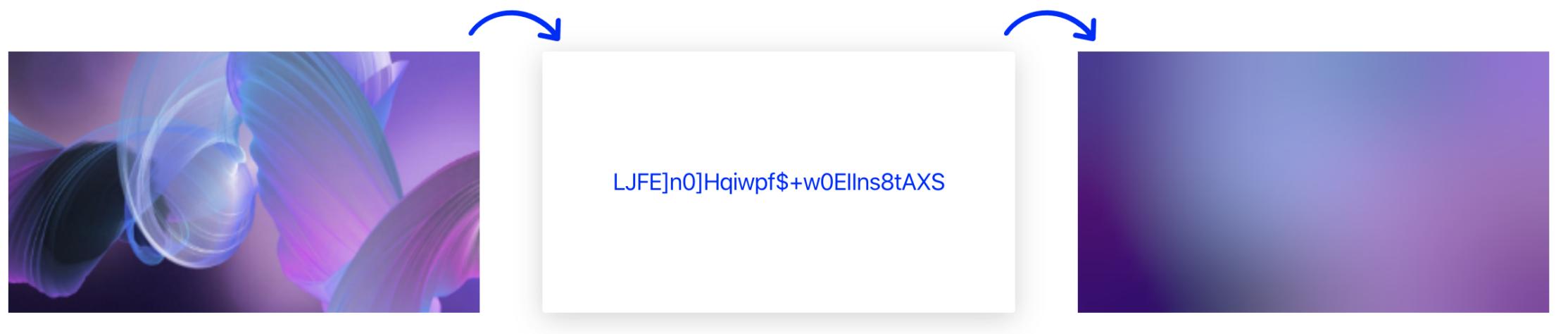


## 4.2. **after\_layout**

Compact representation of a placeholder for an image. Encode a blurry image under 30 characters for instant display like used by Medium

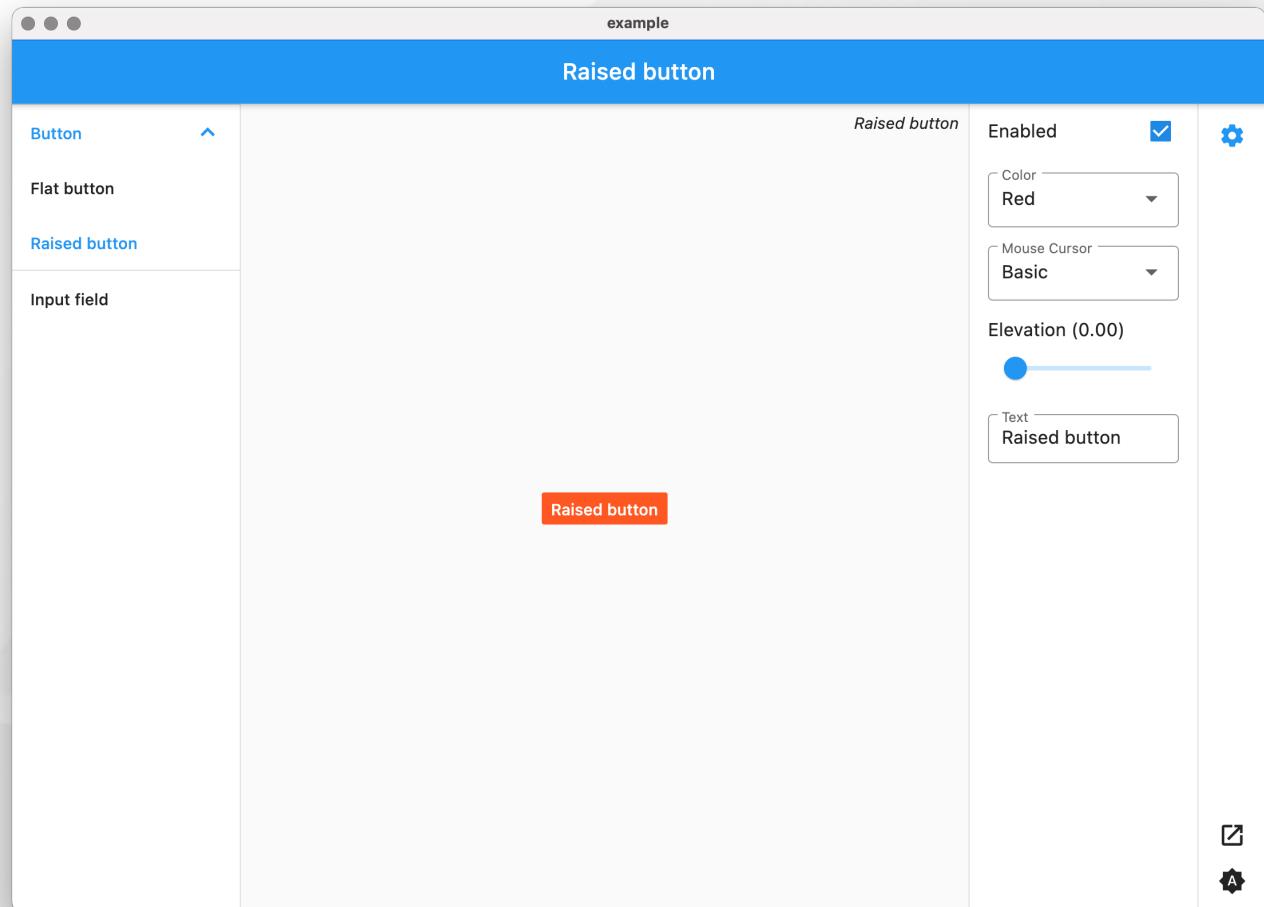
### 4.3. flutter\_blurhash

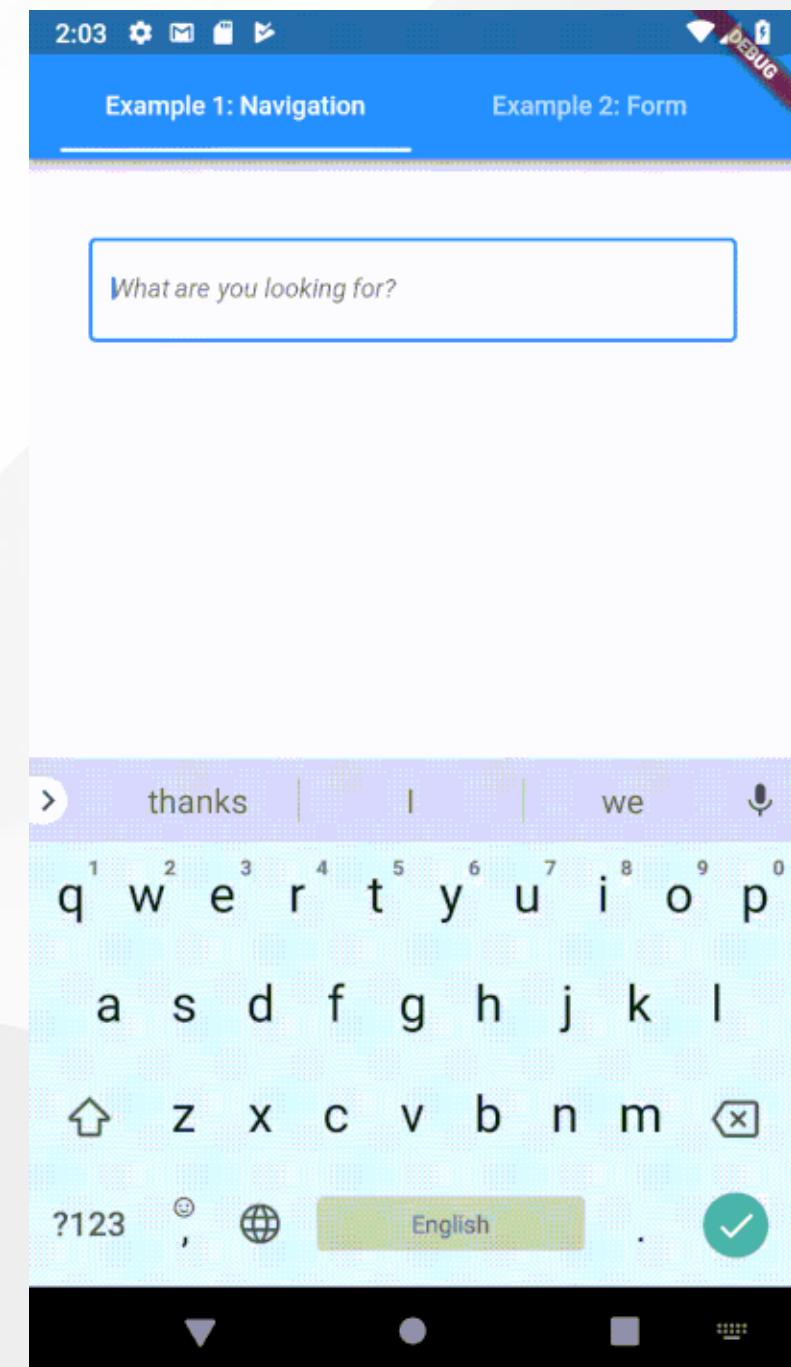
Execute code after the first layout of your widget has been performed, i.e. after the first frame has been displayed.



### 4.4. `storybook_flutter`

A cross-platform storybook for showcasing widgets. It should work in all platforms supported by Flutter.





## 4.1. flutter\_typeahead (1)

A highly customizable typeahead  
(autocomplete) text input field for  
Flutter

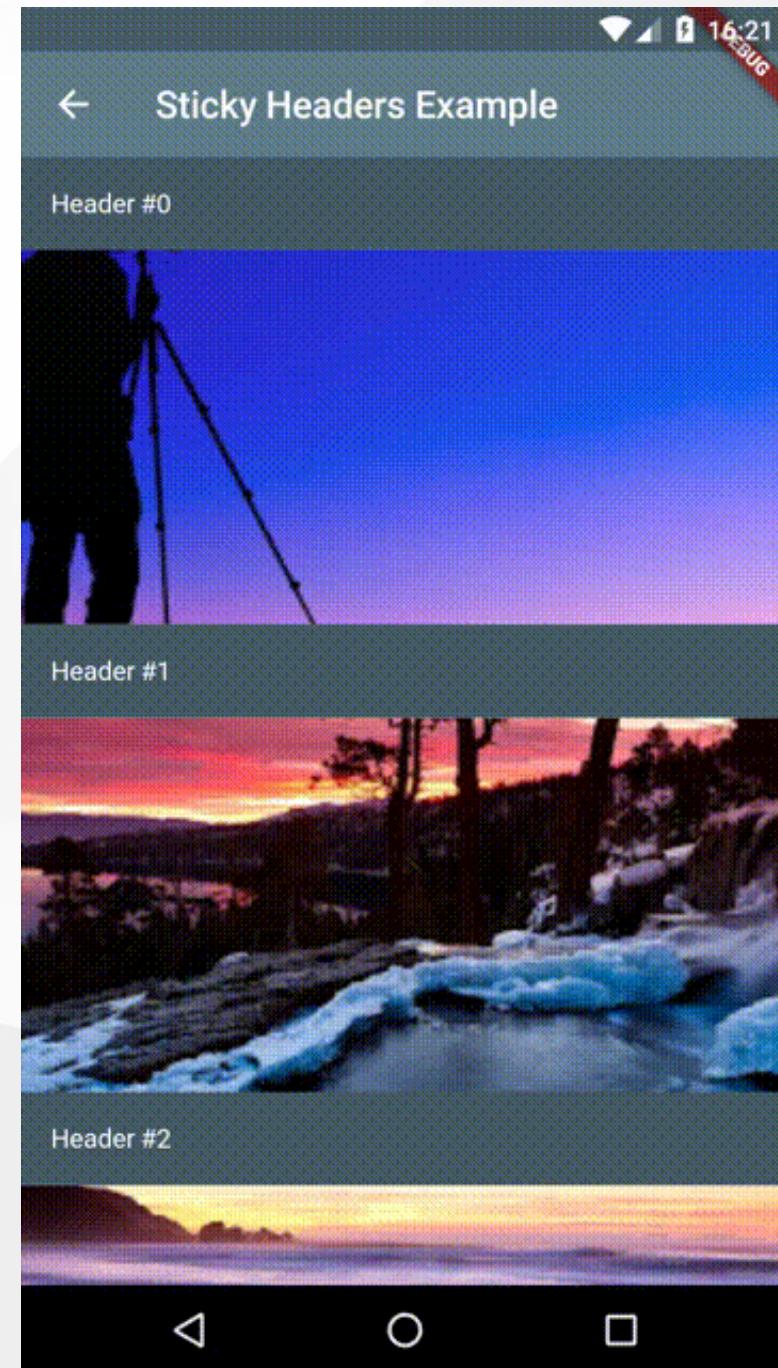
## 4.1. [flutter\\_typeahead](#) (2)

### Features

- Shows suggestions in an overlay that floats on top of other widgets
- Allows you to specify what the suggestions will look like through a builder function
- Allows you to specify what happens when the user taps a suggestion
- Accepts all the parameters that traditional TextFields accept, like decoration, custom TextEditingController, text styling, etc.
- Provides two versions, a normal version and a FormField version that accepts validation, submitting, etc.
- Provides high customizability; you can customize the suggestion box decoration, the loading bar, the animation, the debounce duration, etc.

## 4.1. sticky\_headers

Lets you place headers on scrollable content that will stick to the top of the container whilst the content is scrolled.

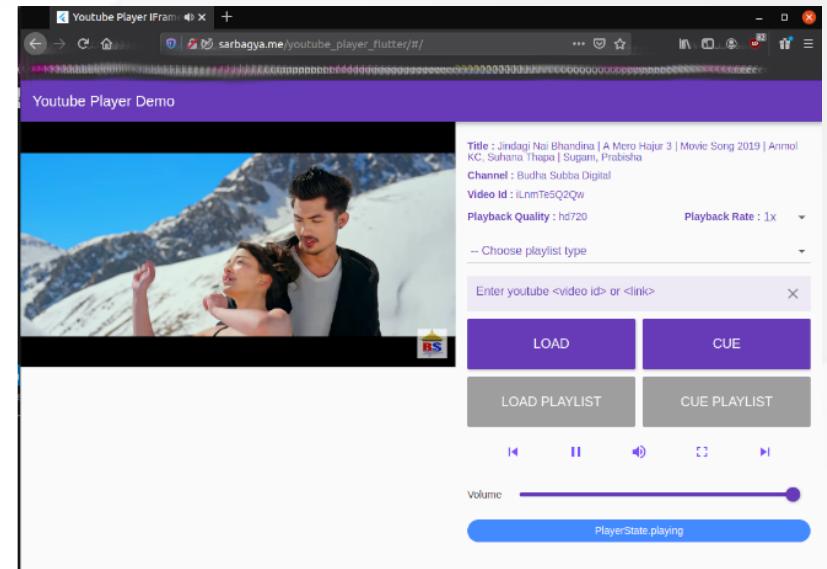
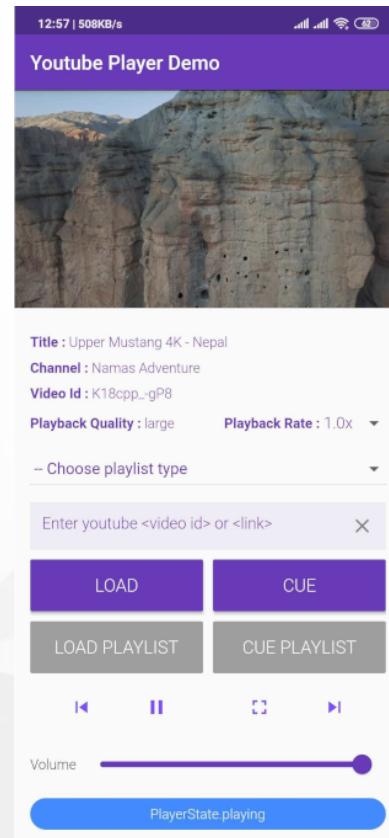


### 4.1. transparent\_image

A transparent image in Dart code, represented as a Uint8List.

### Salient Features

- Inline Playback
- Supports captions
- No need for API Key



## 4.1. youtube\_player\_iframe

### Salient Features

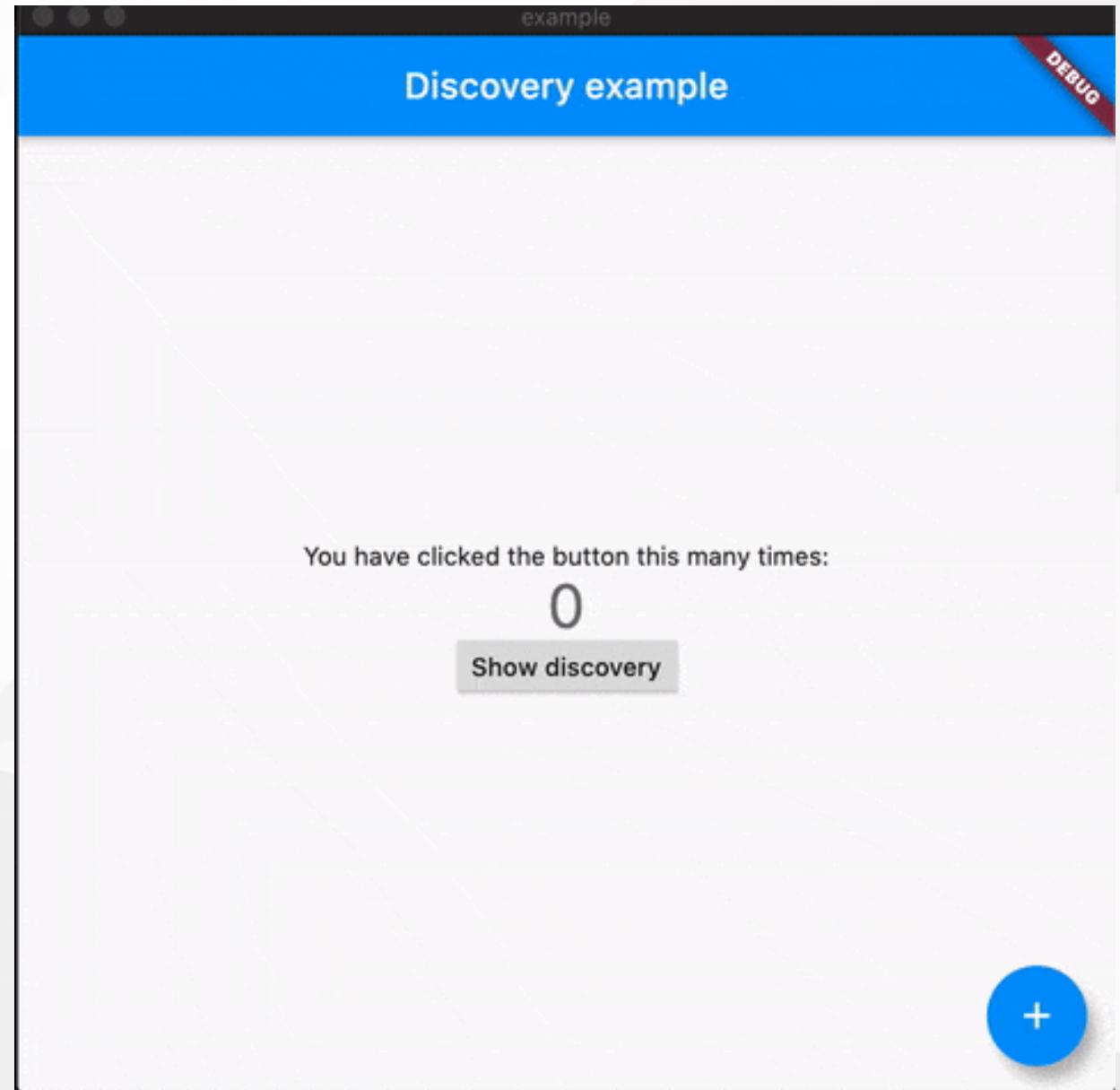
- Supports custom controls
- Retrieves video meta data
- Supports Live Stream videos
- Supports changing playback rate
- Support for both Android and iOS
- Adapts to quality as per the bandwidth
- Exposes builders for building custom controls
- Playlist Support (Both custom and Youtube's playlist)

## 4.1. flutter\_portal

A common use-case for UIs is to show "overlays".

They come in all shapes and forms:

- tooltips
- contextual menus,
- dialogs
- etc



## 4.1. [animated\\_text\\_kit](#) (1)

A flutter package project which contains a collection of cool and beautiful text animations.

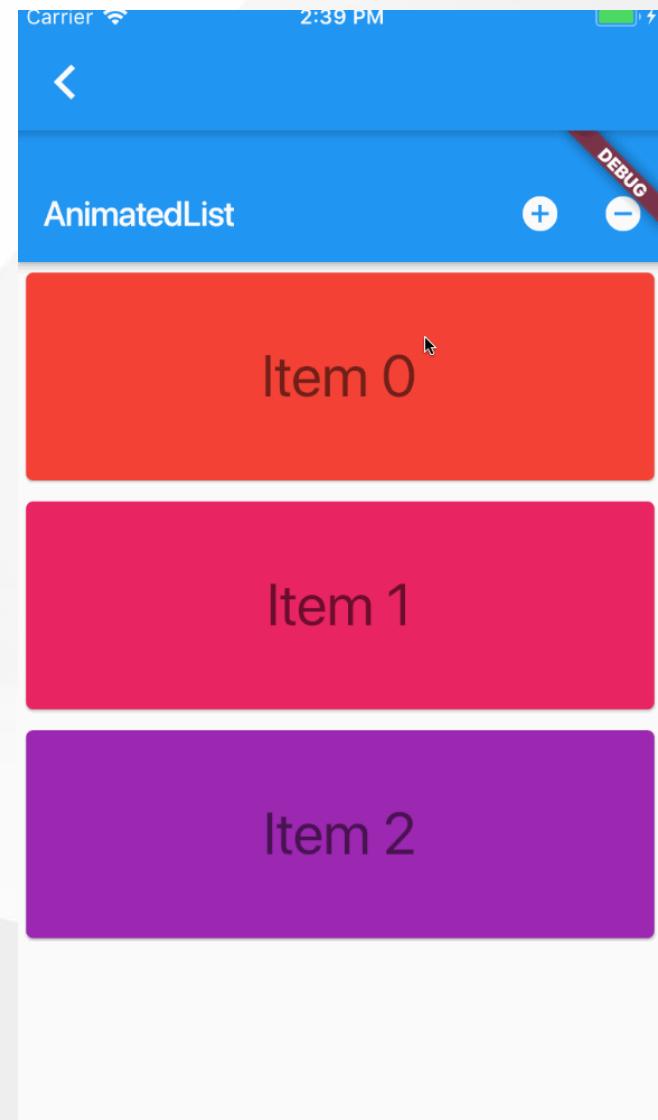


## 4.1. [animated\\_text\\_kit](#) (2)

| Support animations |                              |
|--------------------|------------------------------|
| - Rotate           | - Colorize                   |
| - Fade             | - TextLiquidFill             |
| - Typer            | - Wavy                       |
| - Typewriter       | - Flicker                    |
| - Scale            | - Create your own Animations |

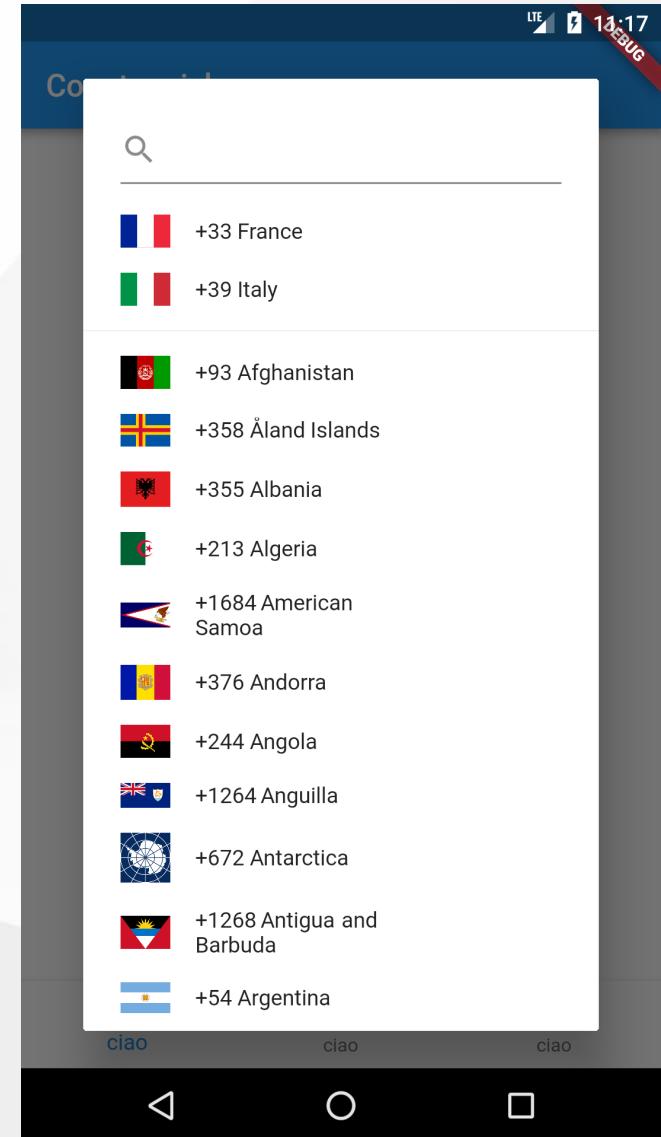
### 4.2. pull\_to\_refresh

A widget provided to the flutter scroll component drop-down refresh and pull up load.



## 4.1. [country\\_code\\_picker](#)

A flutter package for showing a country code selector. In addition it gives the possibility to select a list of favorites countries, as well as to search using a simple searchbox

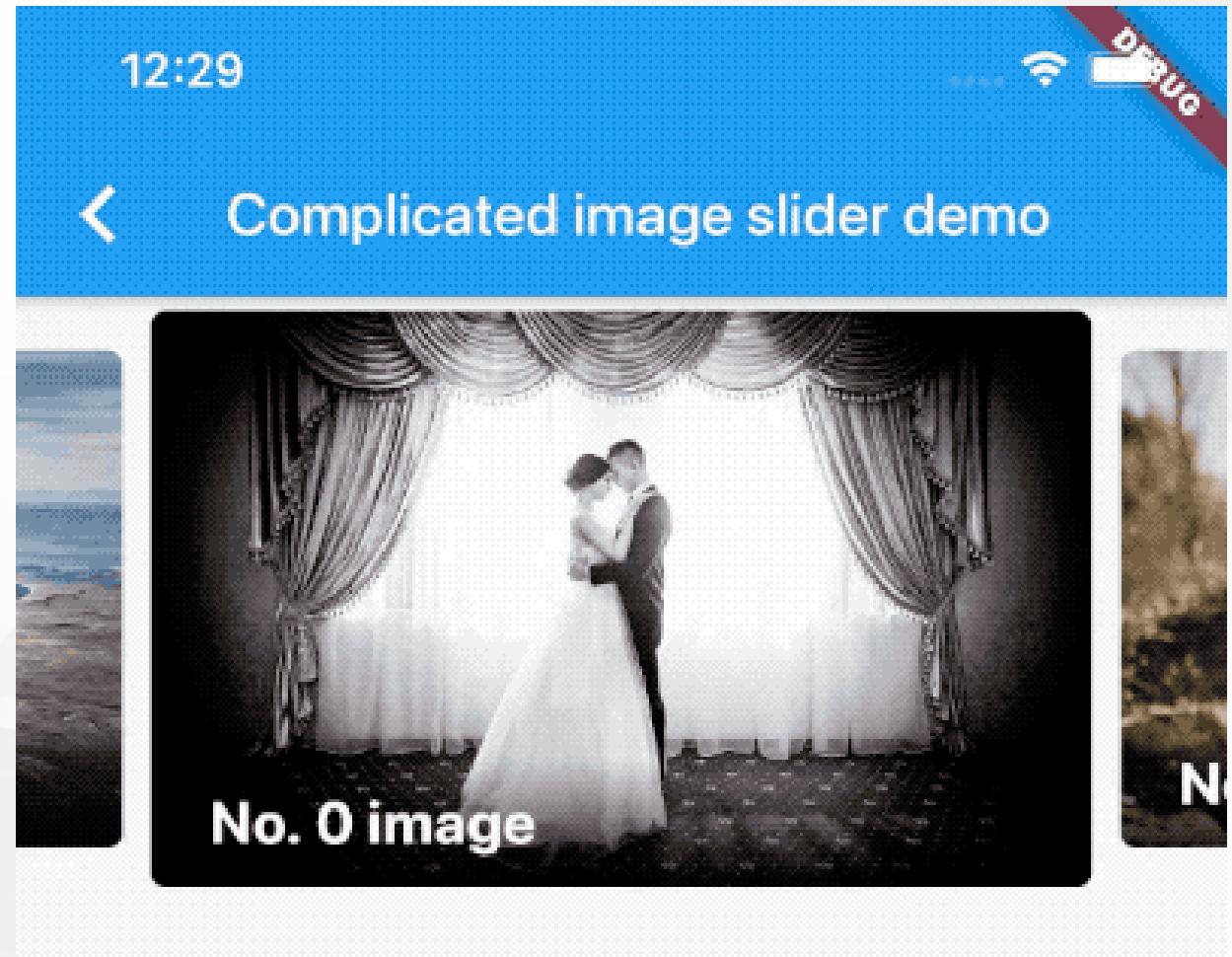


## 4.1. carousel\_slider

A carousel slider widget, support infinite scroll and custom child widget.

### Features

- Infinite scroll
- Custom child widgets
- Auto play



## 4.1. [flutter\\_inappwebview](#)

A Flutter plugin that allows you to add an inline webview, to use an headless webview, and to open an in-app browser window.

### Features

- Infinite scroll
- Custom child widgets
- Auto play



## 5. Data & Network

- logging
- internet\_connection\_checker
- alice\_lightweight
- dio
- pretty\_dio\_logger
- dio\_flutter\_transformer
- retry
- geolocator

## 5.1. logging

Provides APIs for debugging and error logging. This library introduces abstractions similar to those used in other languages, such as the Closure JS Logger and `java.util.logging.Logger`.

## 5.1. [internet\\_connection\\_checker](#)

A pure Dart library that checks for internet by opening a socket to a list of specified addresses, each with individual port and timeout. Defaults are provided for convenience.

### Features

- Login on iOS, Android and Web.
- Express login on Android.
- Granted and declined permissions.
- User information, picture profile and more.
- Provide an access token to make request to the Graph API.

## 5.1. [alice\\_lightweight](#) (1)

Alice is an HTTP Inspector tool for Flutter which helps debugging http requests. It catches and stores http requests and responses, which can be viewed via simple UI. It is inspired from Chuck and Chucker.

Lightweight version of Alice: <https://github.com/jhomlala/alice>

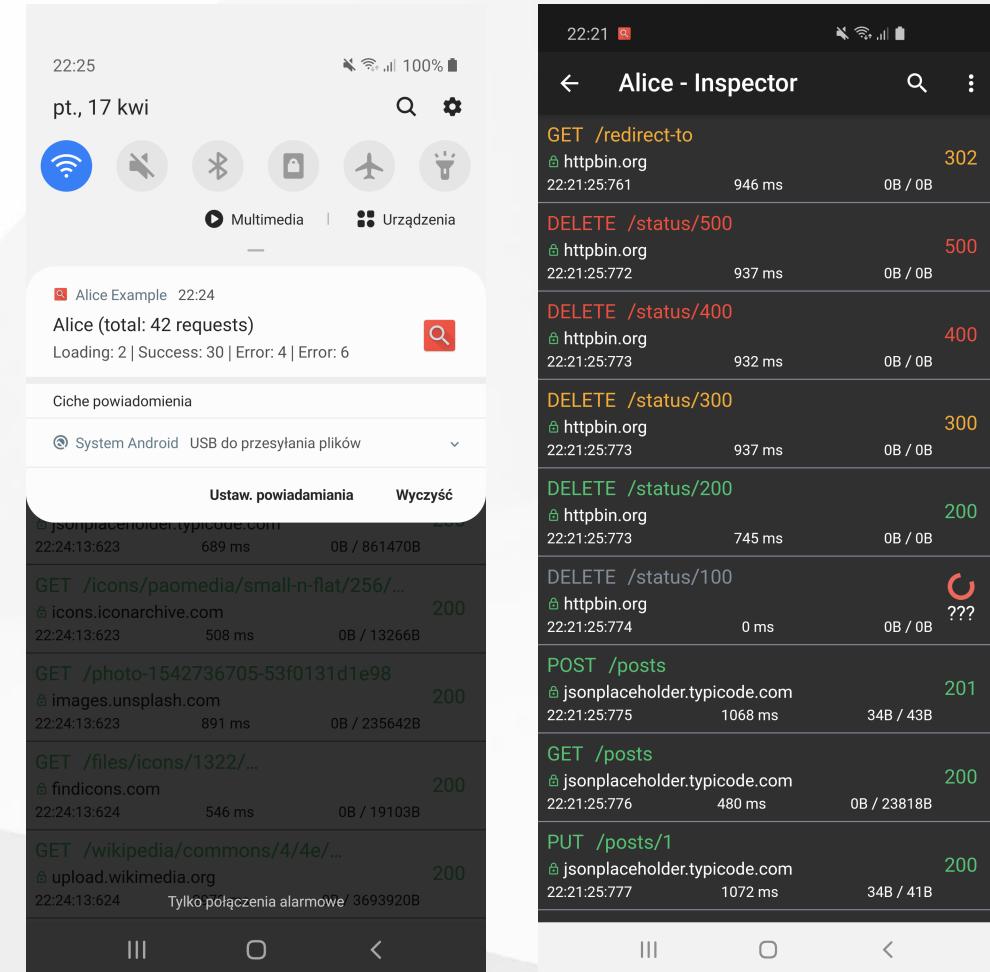
### Supported Dart http client plugins:

- Dio
- HttpClient from dart:io package
- Http from http/http package
- Chopper
- Generic HTTP client

## 5.1. alice\_lightweight (2)

### Features:

- Detailed logs for each HTTP calls (HTTP Request, HTTP Response)
- Inspector UI for viewing HTTP calls
- Save HTTP calls to file
- Statistics
- Notification on HTTP call
- Support for top used HTTP clients in Dart
- Error handling
- Shake to open inspector
- HTTP calls search



## 5.1. dio

Utility for wrapping an asynchronous function in automatic retry logic with exponential back-off, useful when making requests over network.

## 5.1. pretty\_dio\_logger

Pretty Dio logger is a Dio interceptor that logs network calls in a pretty, easy to read format.

## 5.1. dio\_flutter\_transformer

A dio transformer especially for flutter, by which the json decoding will be in background with [compute] function.

## 5.1. **retry**

A powerful Http client for Dart, which supports Interceptors, Global configuration, FormData, Request Cancellation, File downloading, Timeout etc.

## 5.1. geolocator

Geolocation plugin for Flutter. This plugin provides a cross-platform (iOS, Android) API for generic location (GPS etc.) functions.

### Features

- Get the last known location;
- Get the current location of the device;
- Get continuous location updates;
- Check if location services are enabled on the device;
- Calculate the distance (in meters) between two geocoordinates;
- Calculate the bearing between two geocoordinates;

## 6. Firebase and Google services

- google\_mobile\_ads
- firebase\_core
- firebase\_messaging
- cloud\_firestore
- firebase\_performance
- firebase\_crashlytics
- firebase\_dynamic\_links
- firebase\_analytics

## 6.1. [google\\_mobile\\_ads](#)

Flutter plugin for Google Mobile Ads, supporting banner, interstitial (full-screen), rewarded and native ads

## 6.1. **`firebase_core`**

Flutter plugin for Firebase Core, enabling connecting to multiple Firebase apps.

## 6.1. [firebase\\_messaging](#)

Flutter plugin for Firebase Cloud Messaging, a cross-platform messaging solution that lets you reliably deliver messages on Android and iOS.

## 6.1. [cloud\\_firestore](#)

Flutter plugin for Cloud Firestore, a cloud-hosted, noSQL database with live synchronization and offline support on Android and iOS.

## 6.1. [firebase\\_performance](#)

Flutter plugin for Google Performance Monitoring for Firebase, an app measurement solution that monitors traces and HTTP/S network requests on Android and iOS.

## 6.1. [firebase\\_crashlytics](#)

Flutter plugin for Firebase Crashlytics. It reports uncaught errors to the Firebase console.

## 6.1. [firebase\\_dynamic\\_links](#)

Flutter plugin for Google Dynamic Links for Firebase, an app solution for creating and handling links across multiple platforms.

## 6.1. [firebase\\_analytics](#)

Flutter plugin for Google Analytics for Firebase, an app measurement solution that provides insight on app usage and user engagement on Android and iOS.

## 7. Communication with Native and Plugins

- flutter\_secure\_storage
- shared\_preferences
- flutter\_facebook\_auth
- google\_sign\_in
- sign\_in\_with\_apple
- store\_redirect
- url\_launcher
- sms\_autofill
- device\_info
- package\_info

## 7.1. flutter\_secure\_storage

Flutter Secure Storage provides API to store data in secure storage. Keychain is used in iOS, KeyStore based solution is used in Android.

## 7.1. shared\_preferences

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

## 7.1. [flutter\\_facebook\\_auth](#)

The easiest way to add facebook login to your flutter app, get user information, profile picture and more. Web support included.

## 7.1. [google\\_sign\\_in](#)

Flutter plugin for Google Sign-In, a secure authentication system for signing in with a Google account on Android and iOS.

## 7.1. [sign\\_in\\_with\\_apple](#)

Flutter bridge to initiate Sign in with Apple (on iOS, macOS, and Android). Includes support for keychain entries as well as signing in with an Apple ID.

## 7.1. **store\_redirect**

A Flutter plugin to redirect users to an app page in Google Play Store and Apple App Store.

## 7.1. **url\_launcher**

Flutter plugin for launching a URL. Supports web, phone, SMS, and email schemes.

## 7.1. [sms\\_autofill](#)

Flutter plugin for launching a URL. Supports web, phone, SMS, and email schemes.

## 7.1. [device\\_info](#)

Flutter plugin providing detailed information about the device (make, model, etc.), and Android or iOS version the app is running on.

Related to package: [device\\_info\\_plus](#)

## 7.1. [package\\_info](#)

Flutter plugin for querying information about the application package, such as CFBundleVersion on iOS or versionCode on Android.

Related to package: [package\\_info\\_plus](#)

## 8. Packages for development

- flutter\_gen
- flutter\_launcher\_icons
- fvm
- melos

## 8.1. flutter\_gen

The Flutter code generator for your assets, fonts, colors, ... — Get rid of all String-based APIs.

Inspired by SwiftGen.

Related packages:

- [https://pub.dev/packages/flutter\\_gen\\_core](https://pub.dev/packages/flutter_gen_core)
- [https://pub.dev/packages/flutter\\_gen\\_runner](https://pub.dev/packages/flutter_gen_runner)

Tags:

- codegen
- assets

## 8.2. `flutter_launcher_icons`

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

Tags:

- codegen
- assets, launcher icon

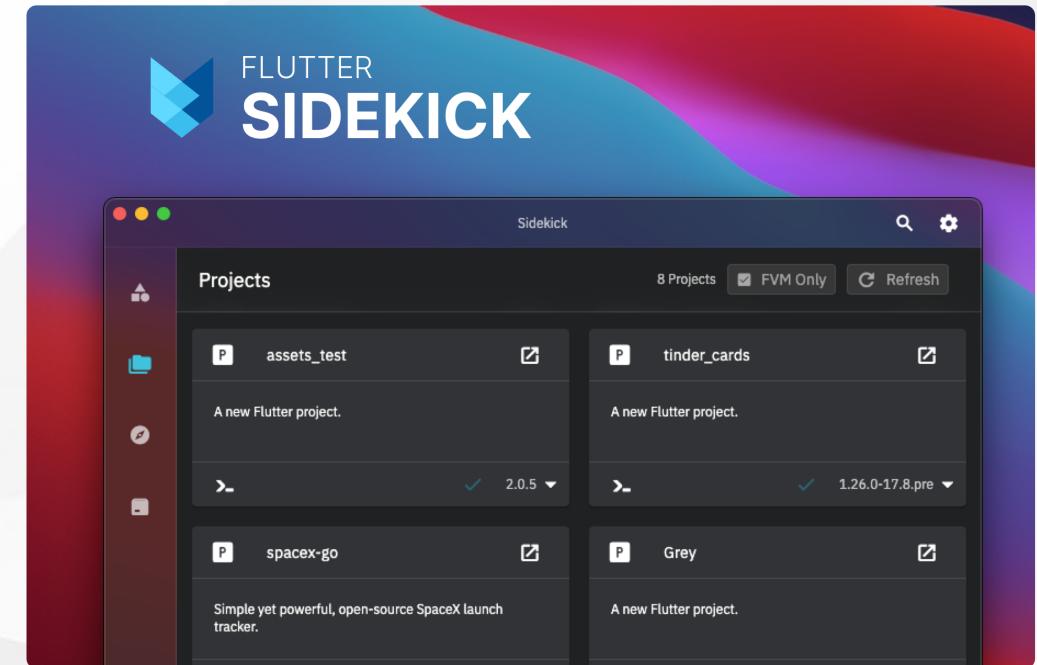
Reference:

- [Flutter: Adding app launcher icons to your project](#) - Dec 8, 2017
- [Home page](#)

### 8.3. `fvm` (1)

A simple cli to manage Flutter SDK versions per project.

Support channels, releases, and local cache for fast switching between versions.



### 8.3. `fvm` (2)

#### Features

- Configure and use Flutter SDK version per project
- Ability to install and cache multiple Flutter SDK Versions
- Fast switch between Flutter channels & versions
- Dynamic SDK paths for IDE debugging support.
- Version FVM config with a project for consistency across teams and CI environments.
- Set global Flutter version across projects

## 8.4. melos

A tool for managing Dart projects with multiple packages.

```
# 1. Install  
pub global activate melos  
  
# 2. Usage  
melos bs  
# melos bootstrap  
  
melos run build:android:staging  
  
melos clean
```



Thank you