

TECHNOLOGY DAY / 13.11.2023

Flutter

Build apps for any screen

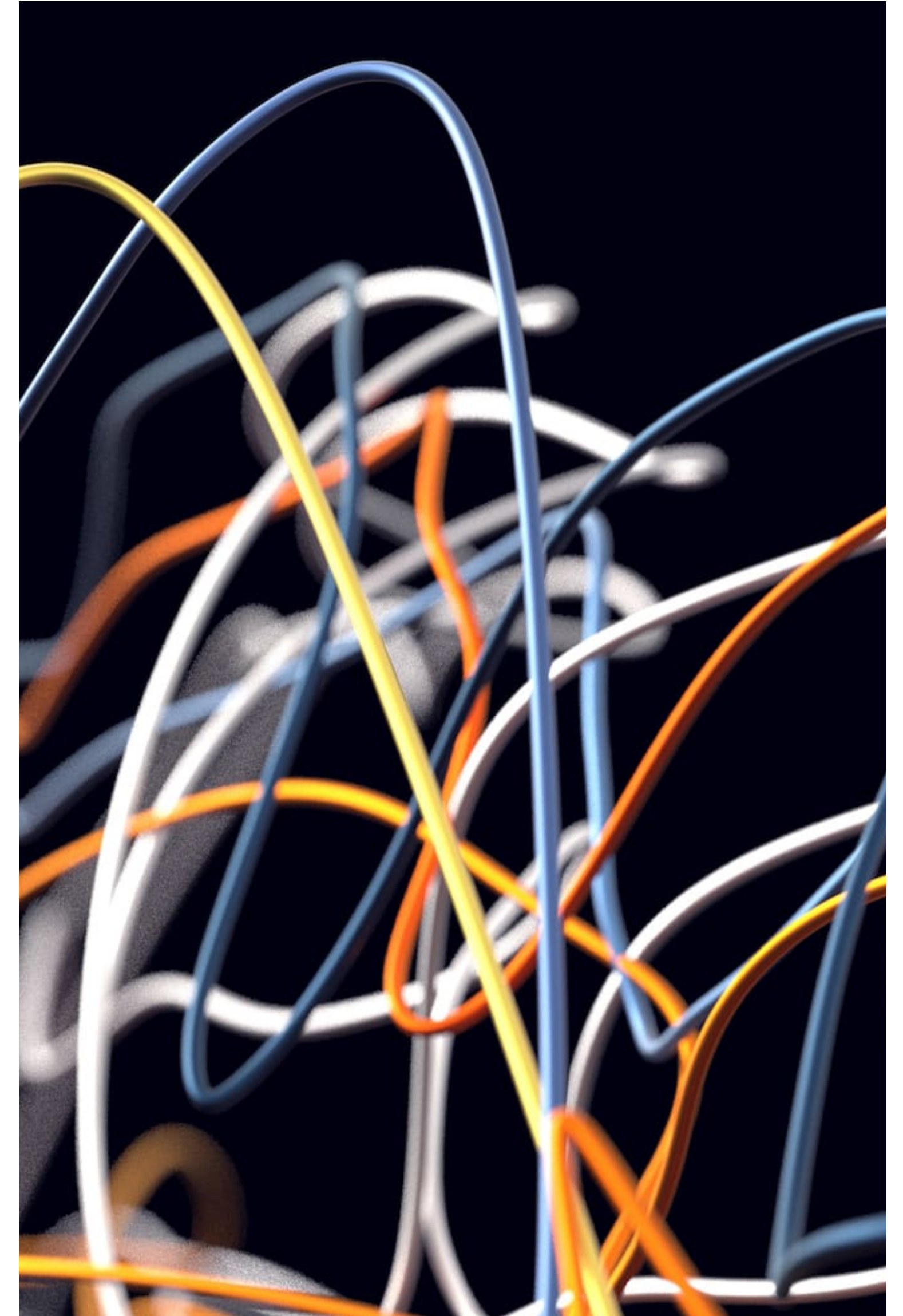
INNOQ



TOBREN KELLER
WERKSTUDENT

Agenda

1. Für mehrere Plattformen Entwickeln
2. Was ist Flutter?
3. Developer Experience
4. Für welche Projekte?
5. Fragen



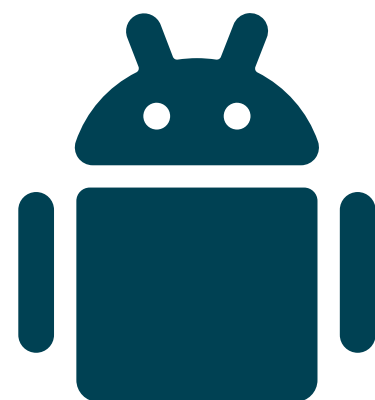


1. Für mehrere Plattformen entwickeln

Verschiedene Technologien



Kotlin



Android



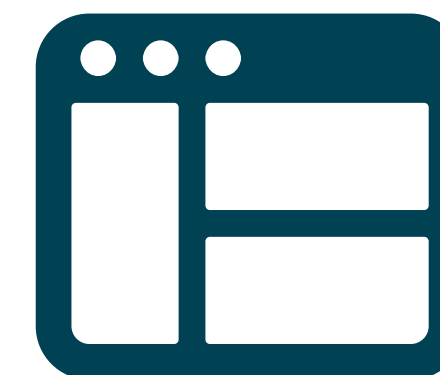
Swift



iOS

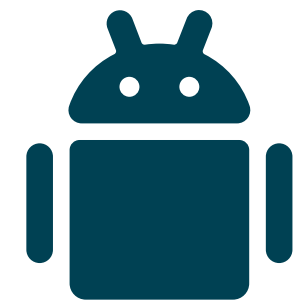


JavaScript



Web

Native Apps sind teuer.



Team Android

Codebase

Backlog

Design



Team IOS

Codebase

Backlog

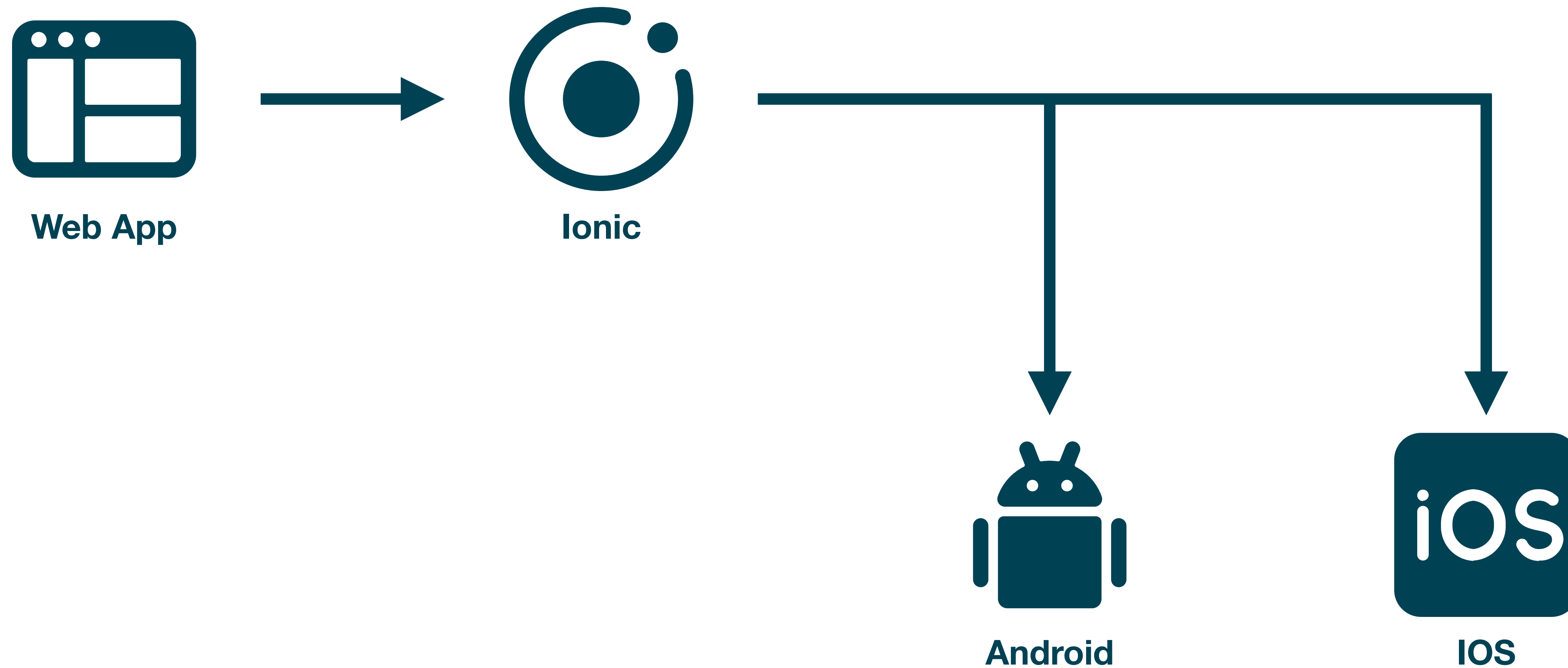
Design



Sehr teuer

Warum nicht einfach Web?

Web zu Mobile



**Performance, Look & Feel
ist nicht wie nativ.**

Flutter will genau das lösen.



2. Was ist Flutter?

Flutter

- Cross-Plattform UI-Framework (Von Google)
- Dart Programmiersprache
- 1.0 seit 2018 (5 Jahre)

- 1 Code -> 7 Plattformen



iOS



Android



Web



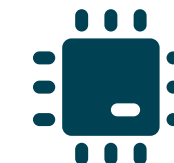
Windows



MacOs



Linux



Embedded

- Native Performance
- Schnelle Produktive Entwicklung



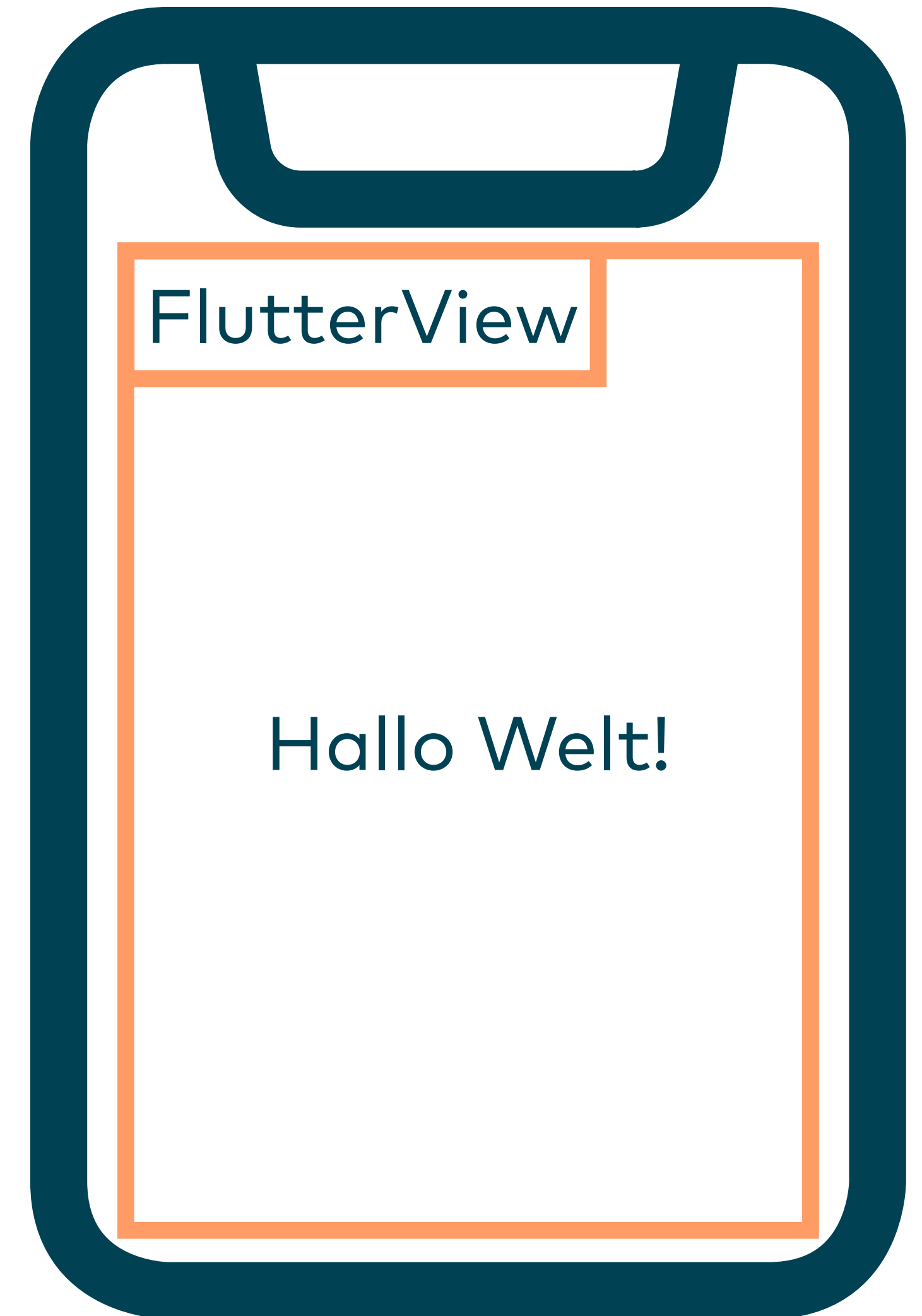
Demo: Wonderous App



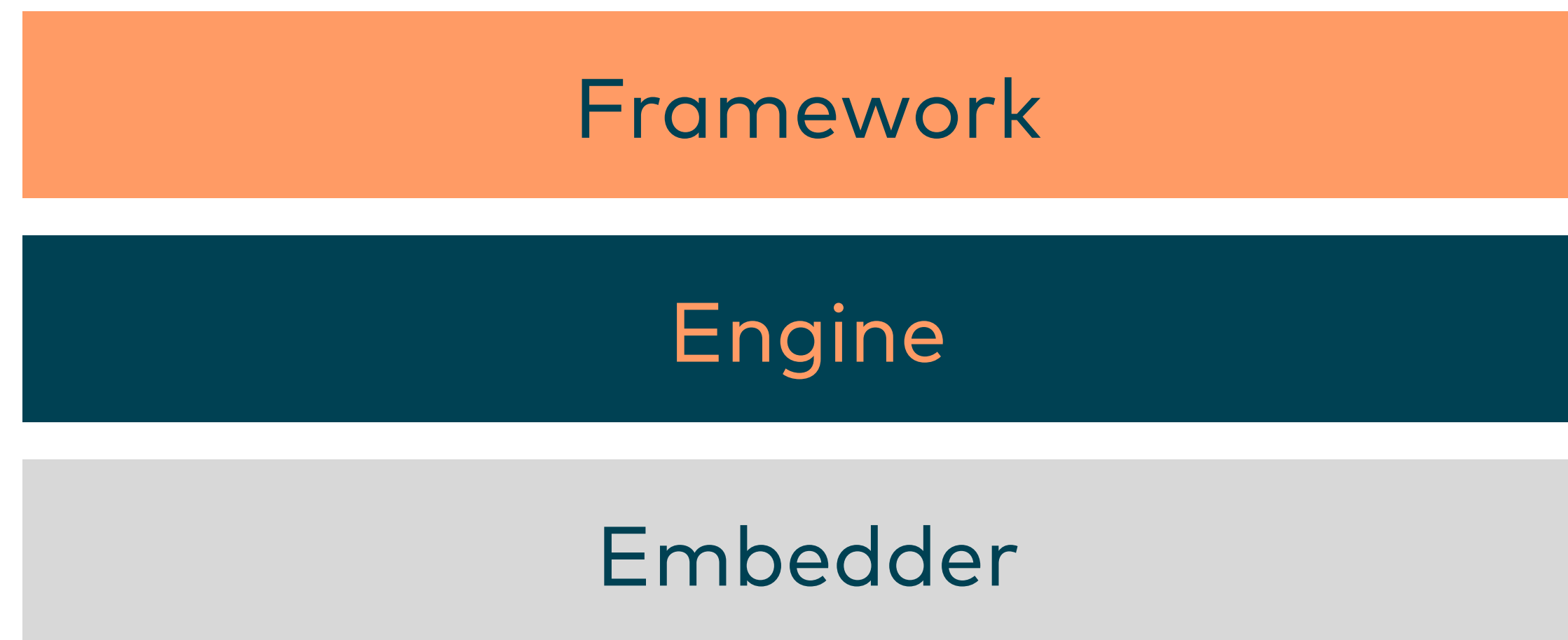
Wie funktioniert Flutter?

Flutter malt selbst.

- FlutterView wird eingebunden
- FlutterView ist leerer Canvas
- Flutter liefert Rendering Engine mit und malt in den Canvas
- Dadurch alle Freiheiten was angezeigt wird
- FlutterView auch in bestehende Apps
- Sogar Einbindung von Fragment Shadern (GLSL) möglich

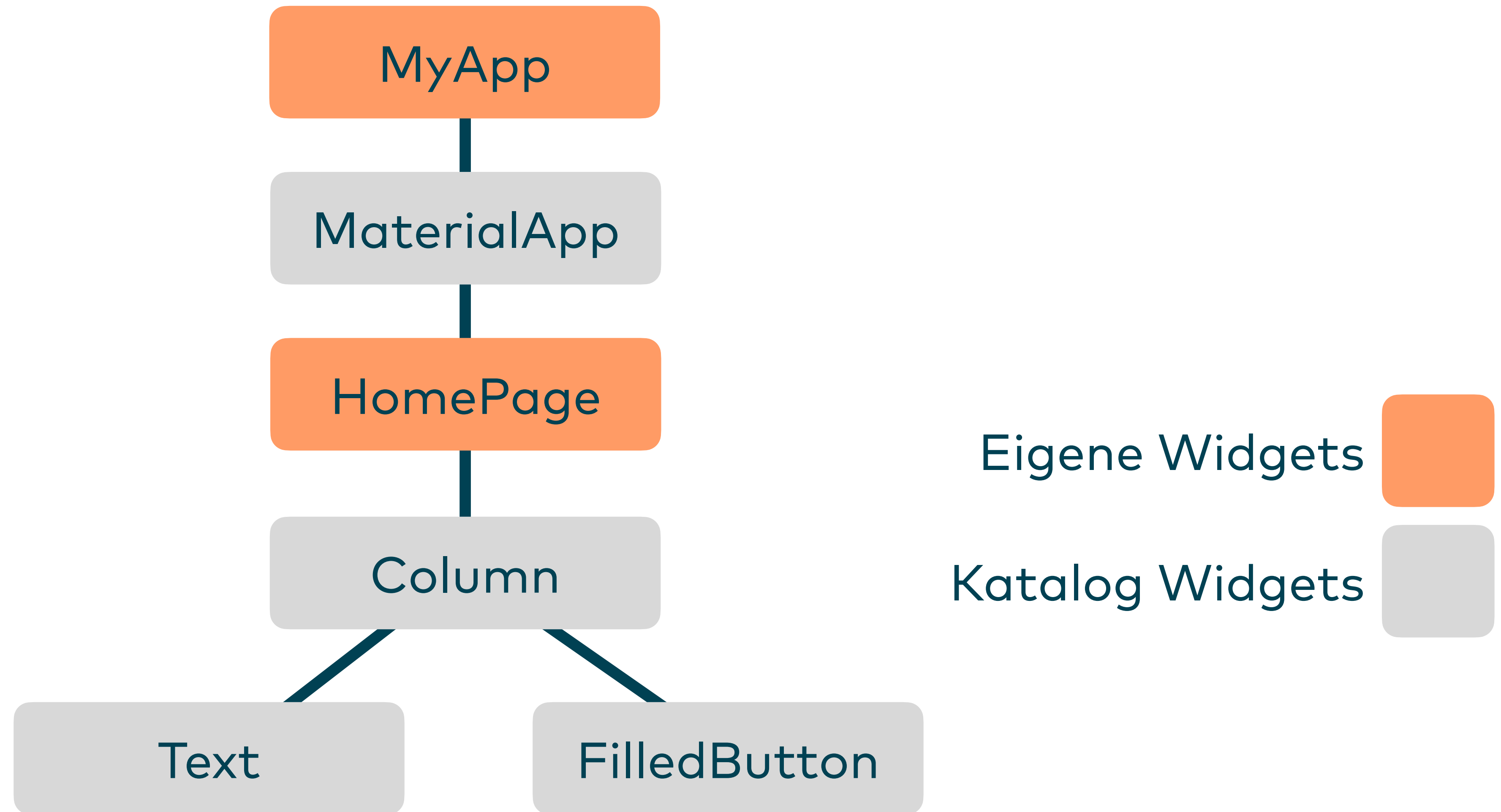


Flutter's Architektur



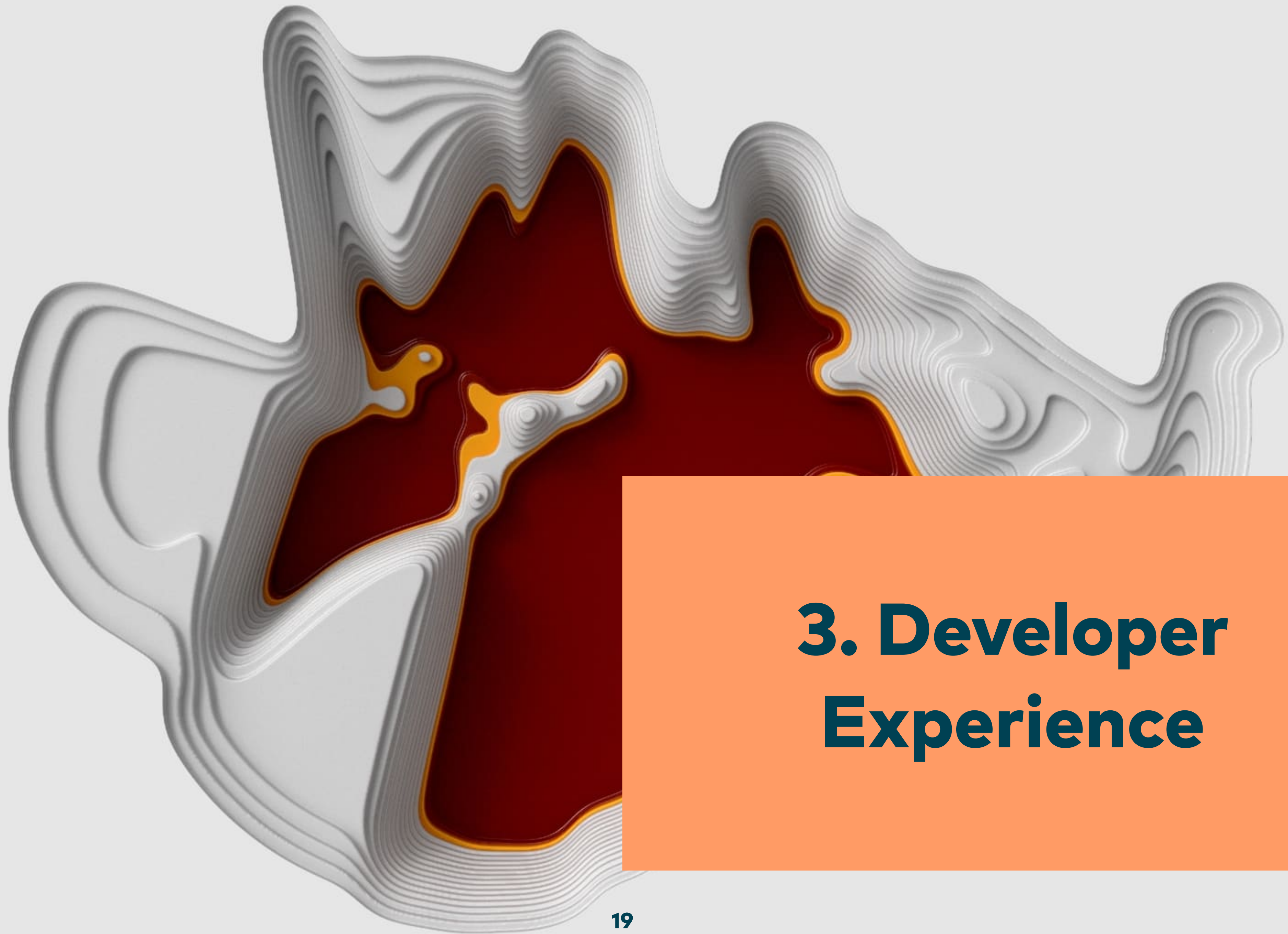
Alles ist ein Widget.

Widget-Tree als Layout



Build Methode erzeugt Widget Tree

```
class MyApp extends StatelessWidget {  
  const MyApp({super.key});  
  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      home: HomePage(),  
    );  
  }  
}
```

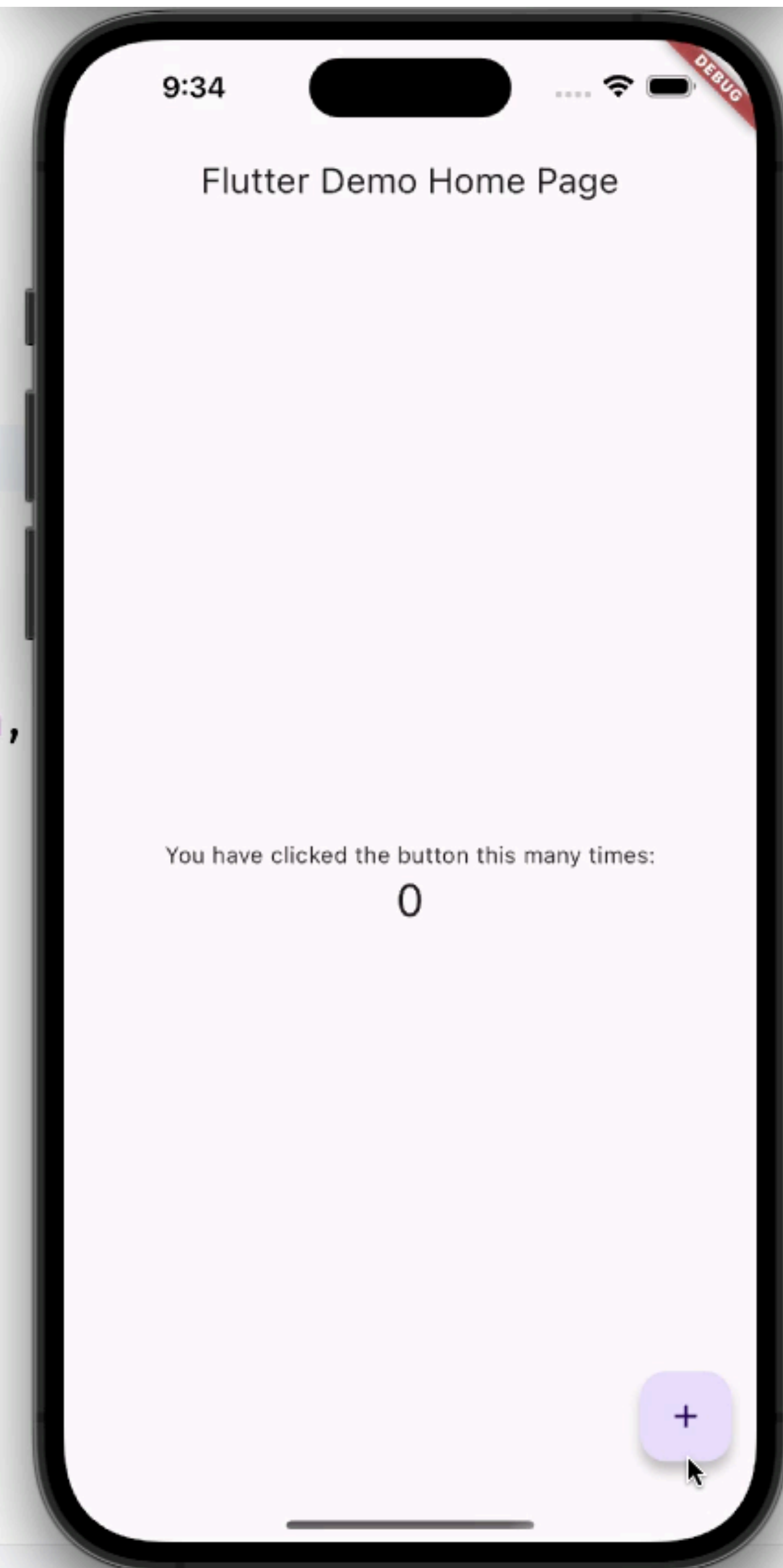
3. Developer Experience

**Hot Reload & Hot Restart
machen Entwicklung super schnell.**


```

65     ), // AppBar
66     body: Center(
67       child: Column(
68         mainAxisAlignment: MainAxisAlignment.center,
69         children: <Widget>[
70           const Text(
71             'You have clicked the button this many times:',
72           ), // Text
73           Text(
74             '$_count',
75             style: Theme.of(context).textTheme.headlineMedium,
76           ), // Text
77         ], // <Widget>[]
78       ), // Column
79     ), // Center
80     floatingActionButton: FloatingActionButton(
81       onPressed: _incrementCounter,
82       tooltip: 'Increment',
83 +   child: const Icon(Icons.add),
84     ), // FloatingActionButton
85   ); // Scaffold
86 }
87 }

```



Flutter ist sehr offen.

- Umfangreicher Widget Katalog
- Material & IOS Widgets im Framework
- Widgets sehr anpassbar
- Flutter + Packages Quelloffen
- Sehr umfangreiche und gute Dokumentation



Flutter ist beliebt.

- 190.000+ Fragen auf StackOverflow
- 158.000+ Sterne auf GitHub
- 1.000.000+ Flutter Apps veröffentlicht
- YouTube Channel mit 500.000+ Abonnenten



Flutter ∅ Dart

Dart macht Spaß.

- Objektorientiert
- Typsicher
- Null-Safe
- async/await + Future
- Wenig Boilerplate
- Eingebautes Linting
- Code Formatting
- DevTools



Flutter / Dart liefern Entwicklertools mit.

Debug Banner

Refresh

Clear

Profile granularity: medium

Performance Overlay

Track Widget Builds

Export



Build - 29.1 ms

CPU Flame Chart

Call Tree

Bottom Up

Expand All

Collapse All

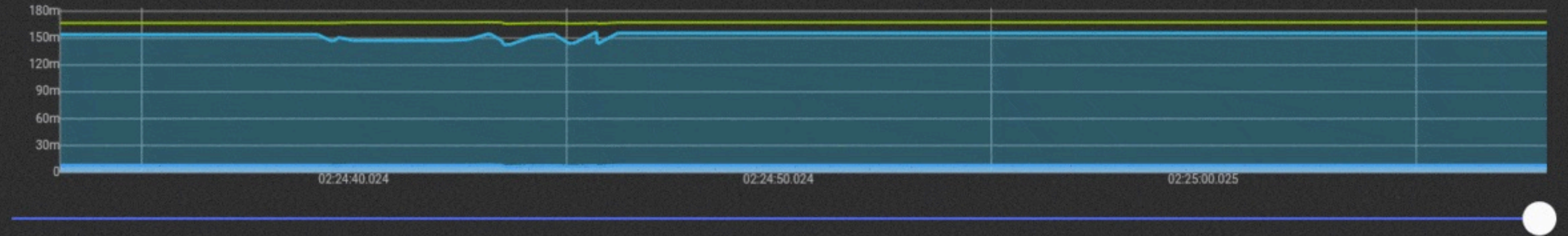
Total Time

Self Time

Method

26.11 ms (90.80%)	0.00 ms (0.00%)	thread_start
26.11 ms (90.80%)	0.00 ms (0.00%)	_pthread_start
0.99 ms (3.45%)	0.00 ms (0.00%)	_pthread_start

Pause Resume Clear Display 1 Minute Source: Live Feed Android Memory Reset GC Export



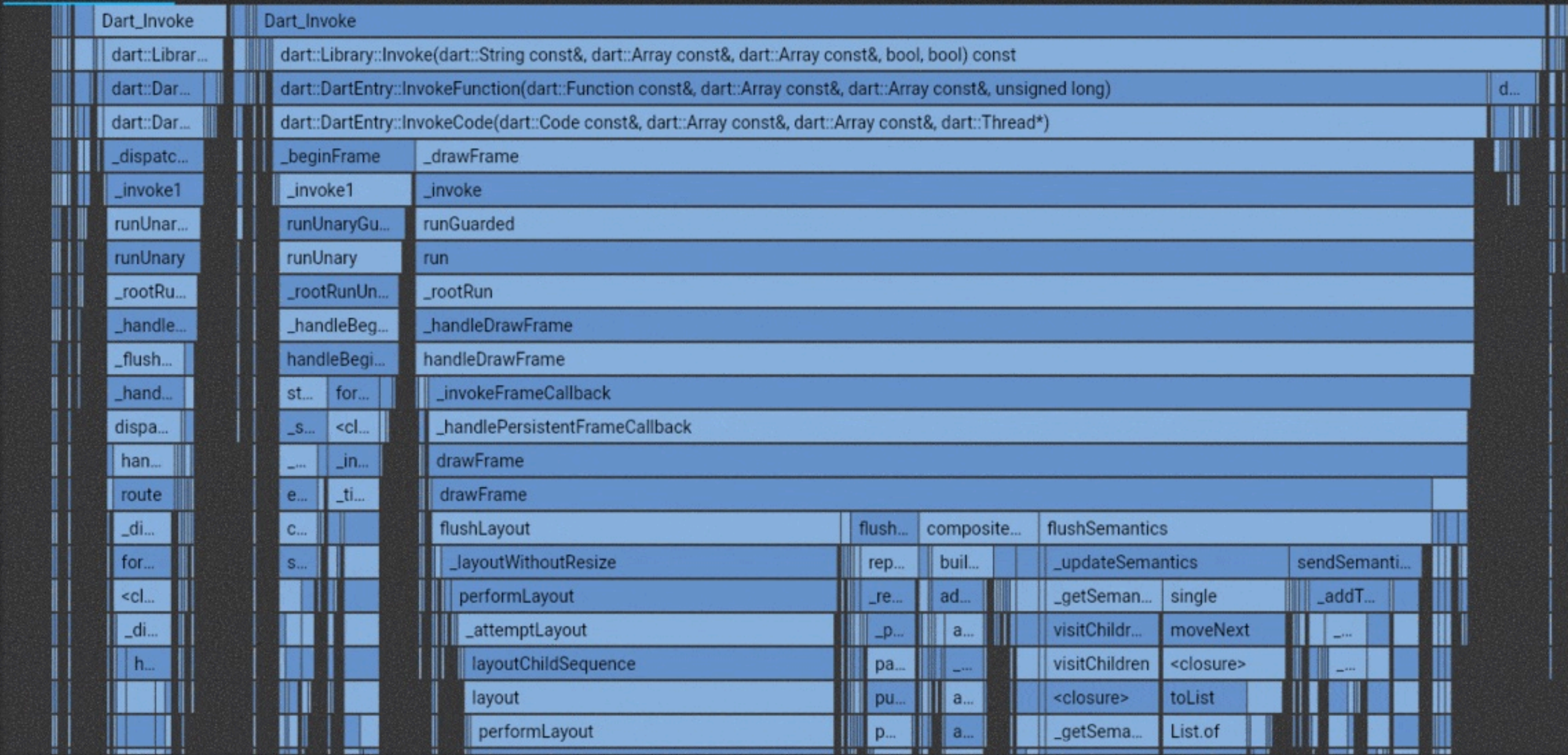
Snapshot Heat Map Group by Library Collapse All Expand All Search Filter Analyze

Library or Class	Count	Shallow	Retained	Type	Name	Value
Analysis	--			CodeSourceMap	code_source_map_	
External	2404	8M	8004912	ExceptionHandler	exception_handlers_	
AutomaticKeepAlive	1	1.1K	1104	Function	owner_	
Code	1	3.01K	3008	Instructions	instructions_	
Instance 0	--	3.18K	176	Instructions	active_instruction_	
CompressedStackMaps	1	3.01K	3008	List	[0]	
Context	2	2.05K	2000	Null	compressed_stackma_	
DebugCreator	2	128	64	ObjectPool	object_pool_	
DiagnosticPropertiesBuilder	1	2K	2000	PcDescriptors	pc_descriptors_	
DiagnosticsProperty	1	2K	2000	_ImmutableList	inlined_id_to_func_	
...	...	...	...	...	...	

Record Stop Clear

Profile granularity: medium Export

CPU Flame Chart Call Tree Bottom Up



☒ Record HTTP traffic☐ Stop Clear

Method	Request URI	Status	Duration	Timestamp
GET	http://www.example.com/	200	50 ms	2:30:09.005 PM
GET	http://www.example.com/	200	56 ms	2:30:19.006 PM
GET	http://www.example.com/	200	65 ms	2:30:29.006 PM
GET	http://www.example.com/	200	54 ms	2:30:39.003 PM
GET	http://www.example.com/	200	47 ms	2:30:49.006 PM
GET	http://www.example.com/	200	53 ms	2:30:59.007 PM
GET	http://www.example.com/	200	57 ms	2:31:09.007 PM
GET	http://www.example.com/	200	49 ms	2:31:19.004 PM
GET	http://www.example.com/	200	60 ms	2:31:29.003 PM
GET	http://www.example.com/	200	47 ms	2:31:39.004 PM
GET	http://www.example.com/	200	52 ms	2:31:49.008 PM
GET	http://www.example.com/	200	52 ms	2:31:59.005 PM
GET	http://www.example.com/	200	52 ms	2:32:09.006 PM
GET	http://www.example.com/	200	54 ms	2:32:19.006 PM
GET	http://www.example.com/	200	52 ms	2:32:29.003 PM
GET	http://www.example.com/	200	48 ms	2:32:39.005 PM
GET	http://www.example.com/	200	52 ms	2:32:49.005 PM
GET	http://www.example.com/	200	57 ms	2:32:59.006 PM
GET	http://www.example.com/	200	60 ms	2:33:09.005 PM
GET	http://www.example.com/	200	55 ms	2:33:19.005 PM
GET	http://www.example.com/	200	78 ms	2:33:29.006 PM

Headers

Timing

General

method: GET

uri: http://www.example.com/

isolateId: isolates/76796289279471

isolateGroupId: isolateGroups/16860335164400473611

compressionState: HttpClientResponseCompressionState.decompressed

connectionInfo: {localPort: 53877, remoteAddress: 2606:2800:220:1:248:1893:25c8:1946, remotePort: 80}

contentLength: 648

cookies: []

isRedirect: false

persistentConnection: true

reasonPhrase: OK

redirects: []

statusCode: 200

Response Headers

last-modified: [Thu, 17 Oct 2019 07:18:26 GMT]

cache-control: [max-age=604800]

date: [Wed, 24 Jun 2020 21:30:18 GMT]

vary: [Accept-Encoding]

Call Stack

main.<closure>() main.dart 25

_rootRunUnary() zone.dart 1198

_rootRunUnary() zone.dart

_CustomZone.runUnary() zone.dart 1100

_CustomZone.runUnaryGuarded() zone.dart 1005

_CustomZone.bindUnaryCallbackGuarded.<clos...

_rootRunUnary() zone.dart 1206

_rootRunUnary() zone.dart

CustomZone.runUnary() zone.dart 1100

Variables

client: _HttpClient

_closing: false

_closingForcefully: false

_connectionTargets: _HashMap

_elementCount: 1

_buckets: _List (8 items)

_modificationCount: 1

_credentials: _GrowableList (0 items)

proxvCredentials: GrowableList (0 items)

Breakpoints

3

• main.dart:25 (package:performance_test_app/...

• main.dart:30 (package:performance_test_app/...

• main.dart:38 (package:performance_test_app/...

Pause

Resume

Step In

Step Over

Step Out

Ignore

Libraries

<

>

package:performance_test_app/main.dart

```
14 import 'package:performance_test_app/src/lime.dart';
15 import 'package:performance_test_app/src/orange.dart';
16 import 'package:performance_test_app/src/pink.dart';
17 import 'package:performance_test_app/src/welcome.dart';
18 import 'package:performance_test_app/src/yellow.dart';
19
20 void main() {
21   // Just something to have in the networking tab and logging tab.
22   HttpClient client = HttpClient();
23   Timer.periodic(const Duration(seconds: 10), (_) async {
24     const networkLoggerName = 'my_network_log';
25     developer.log('Sending request to example.com.', name: ...
26     var request = await client.getUrl(
27       Uri.parse('http://www.example.com/'),
28     );
29     var response = await request.close();
30     developer.log(
31       'Response from example.com received (HTTP $_
32       name: networkLoggerName,
33       error: '{"contentLength": ${response.contentLength}}',
34     );
35   });
36 }
```

Libraries

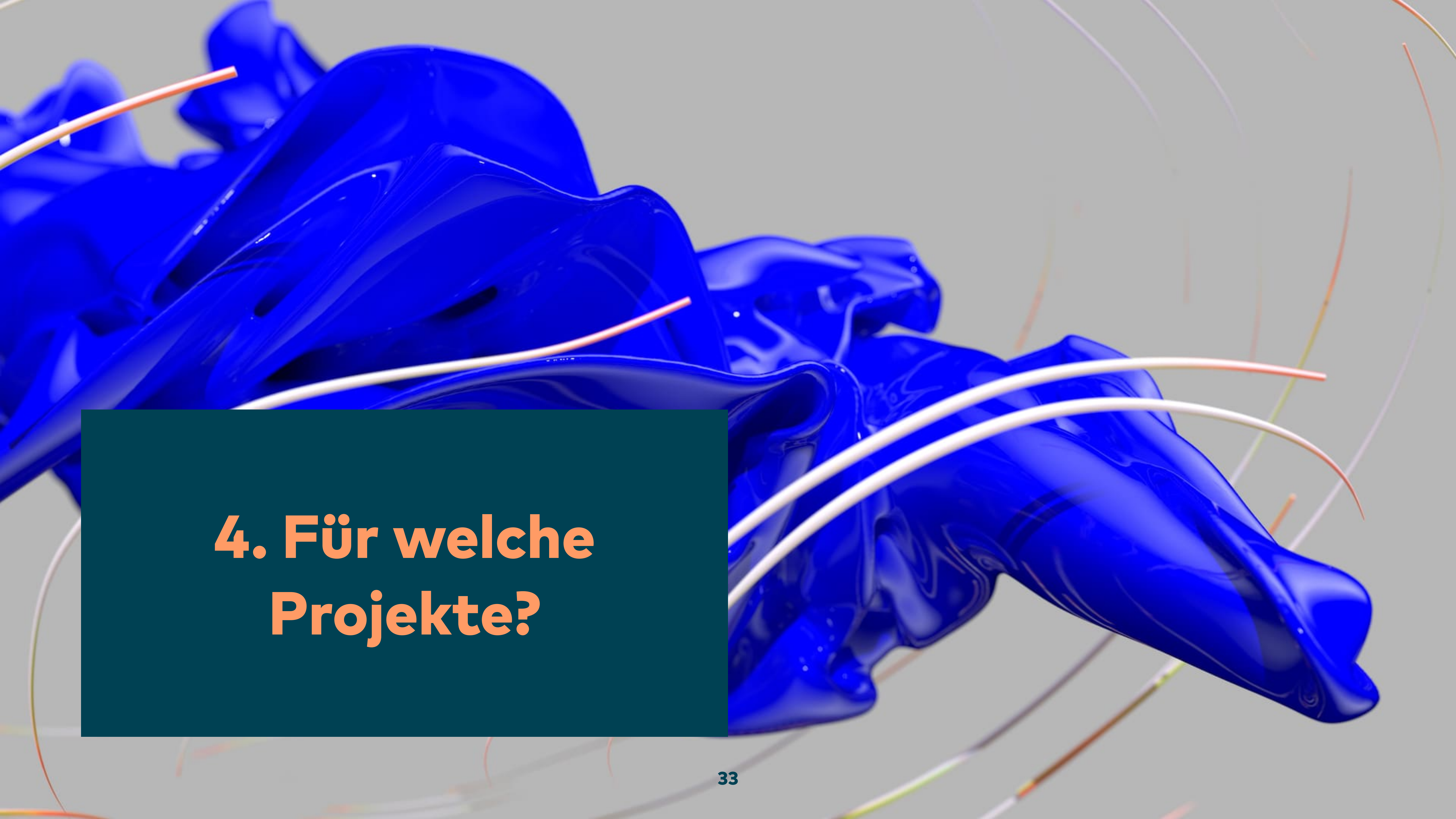
802

Filter (kP)

- dart:async
- dart:async-patch/async_patch.dart
- dart:async-patch/...
- dart:async-patch/...
- dart:async-patch/timer_patch.dart
- dart:async/async_error.dart
- dart:async/...
- dart:async/deferred_load.dart
- dart:async/future.dart
- dart:async/future_impl.dart
- dart:async/schedule_microtask.dart
- dart:async/stream.dart
- dart:async/stream_controller.dart
- dart:async/stream_impl.dart
- dart:async/stream_pipe.dart

Console

```
flutter: Regular print() message.
flutter: Regular print() message.
flutter: Regular print() message.
flutter: Regular print() message.
flutter: Regular print() message.
flutter: Regular print() message.
flutter: Regular print() message.
```

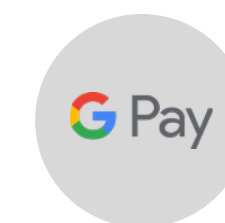



4. Für welche Projekte?

Flutter & neue mobile Apps

Flutter bei Google

- Google Pay App nativ für IOS und Android
- Rewrite in Flutter
- 35% weniger Code
- 10% native Entwickler, 90% Flutter Entwickler
 - > 45% weniger Entwickler
- $\approx$ 60% - 70% Zeitersparnis
 - > Flutter leichter zu programmieren

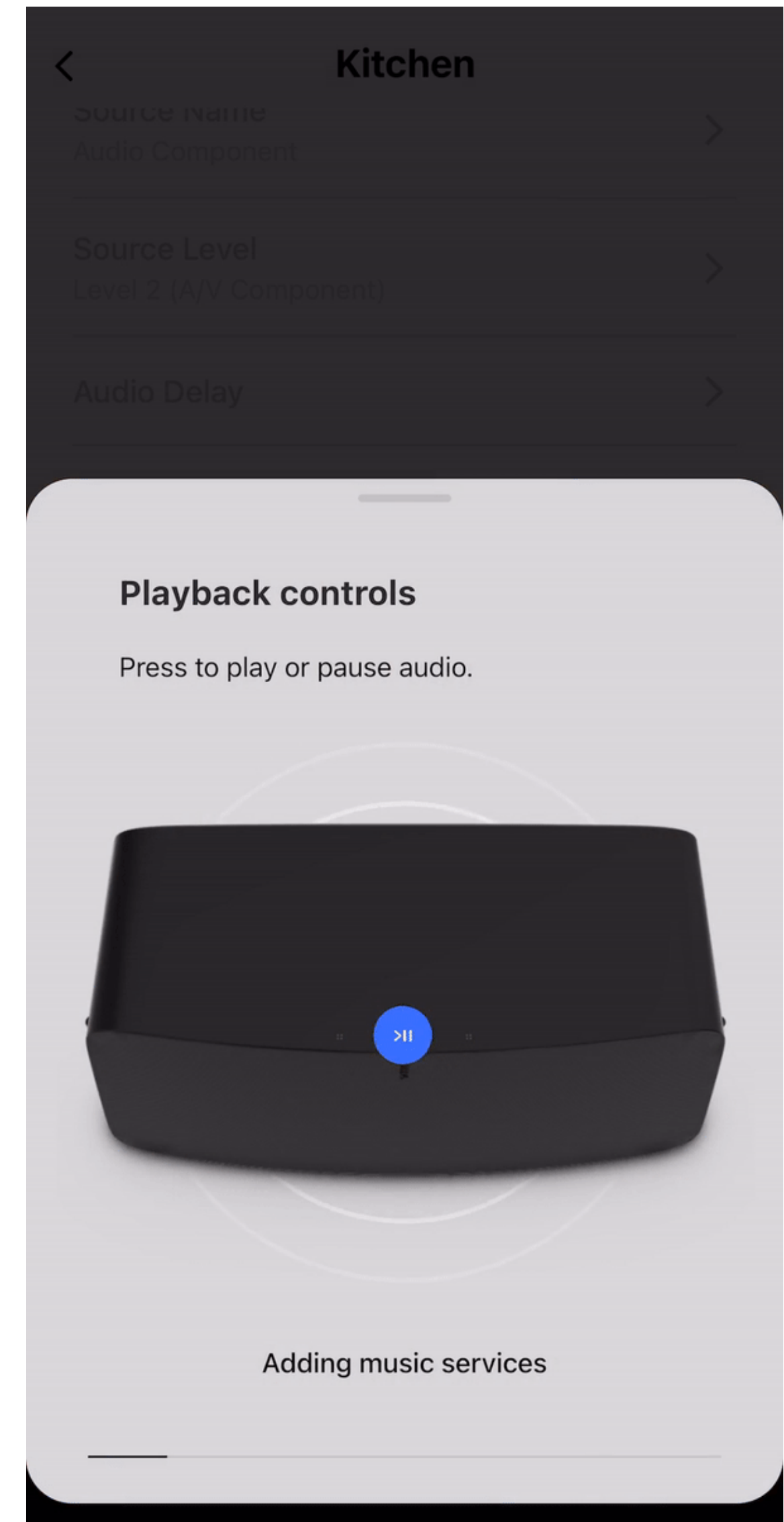


FLUTTER SHOWCASE
GOOGLE PAY

Flutter & bestehende mobile Apps

Flutter bei Sonos

- 2018: Neue UI-Strategie bei Sonos: "show, don't tell"
- 2 native mobile Apps
- Integration von Flutter in bestehende Apps



"We were sold on Flutter's visuals, and can happily attest we were not one bit oversold. Implementing our designers' vision was finally possible—and on more than a few occasions, developers have been able to suggest solutions designers wouldn't have dreamed possible."



"[...] we didn't anticipate how much our automation team would take a liking to Flutter as a tool for building great UIs. Beyond using Flutter to automate wizard flows, they've built a desktop tool (yes, Flutter runs on desktop!) for testers and automation alike."



Kompromisse

- Updates von nativen Komponenten müssen "nachgebaut" werden
- Dart
 - > wenige Entwickler
- "Große" App-Größe
 - > min. 4MB vs. 550kb Native
- viele Framework & Language updates
- Abhängigkeit von Google



Flutter Mobile

- Super performance
- Look & Feel wie nativ
- Nutzung aller Sensoren möglich
- Beschleunigt Entwicklung
- Günstiger als native Apps
- Sollte man ernsthaft in Erwägung ziehen



Flutter & Web

Flutter bei Rive

- Tool zum erstellen von Animationen
- Vorher Web Technologien, jetzt Flutter
- Zuerst Flutter Web App
- Jetzt auch Desktop
- Performance super



New File

Beta 0.8.303

New Artboard - State Machine 1

Previewing a copy of Langos58's file

100%

Design

Animate

New Artboard

Id

ly

lx

r eye

l eye

shadow 1

shadow 2

make up

eye shadows

face

ears

shirt

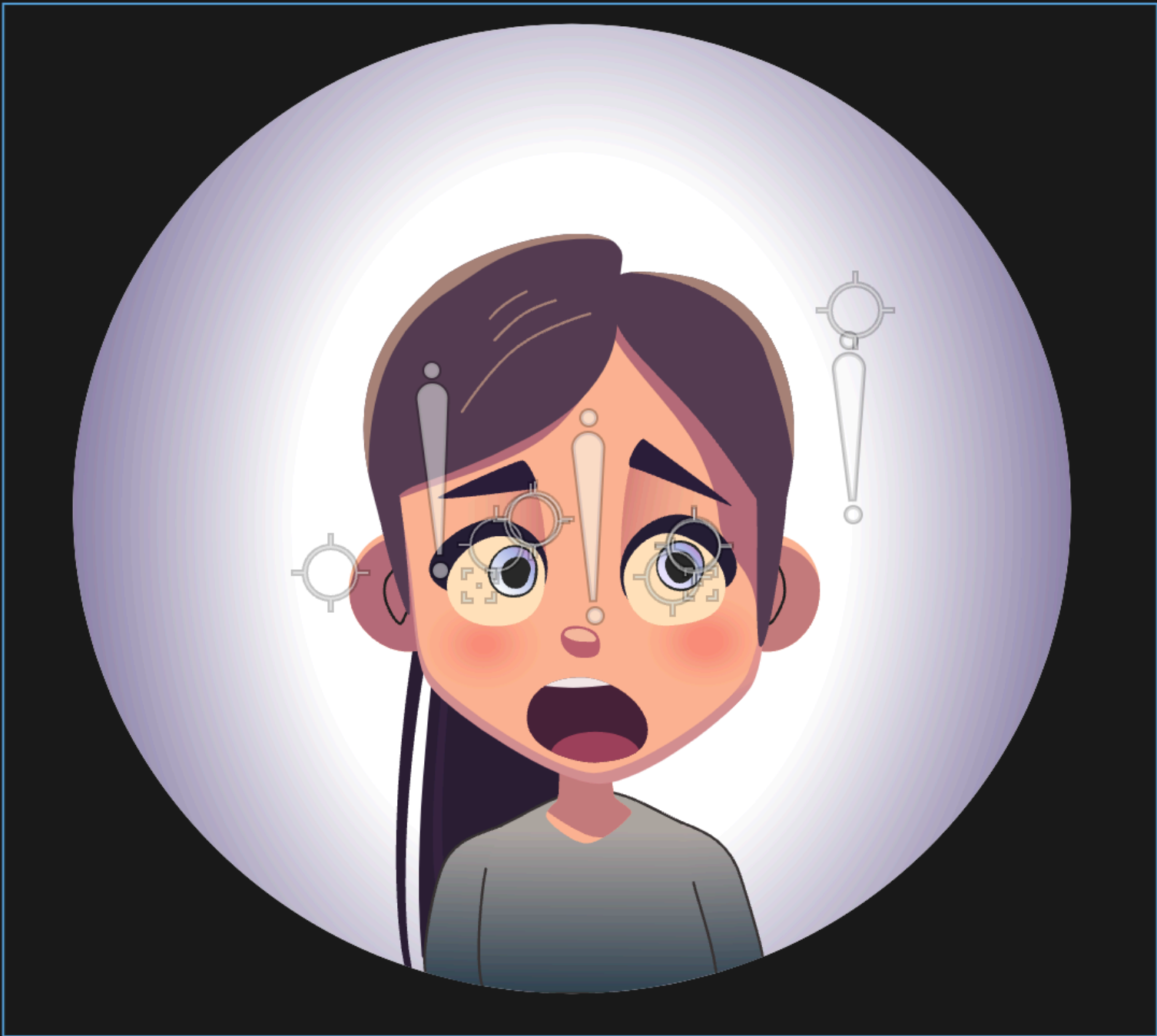
mask

mask

Group

ANIMATING

New Artboard



Transform

Position

-463

-118

X

Y

Size

535.7

479.1

Width

Height

Origin

0%

0%

X

Y

Clip

Fills

Fill 2

1D1D1D

100%

Fill 1

35495D

100%

Strokes

Selection Colors

A78278

100%

383535

100%

212020

100%

Transitions

Timeline 1

Animations

Timeline 2

State Machine 1

Timeline 1

Inputs

Listeners

92%

Layer 1

Layer 2

Entry

→ Timeline 1

Any State

Exit

Console

Flutter Web

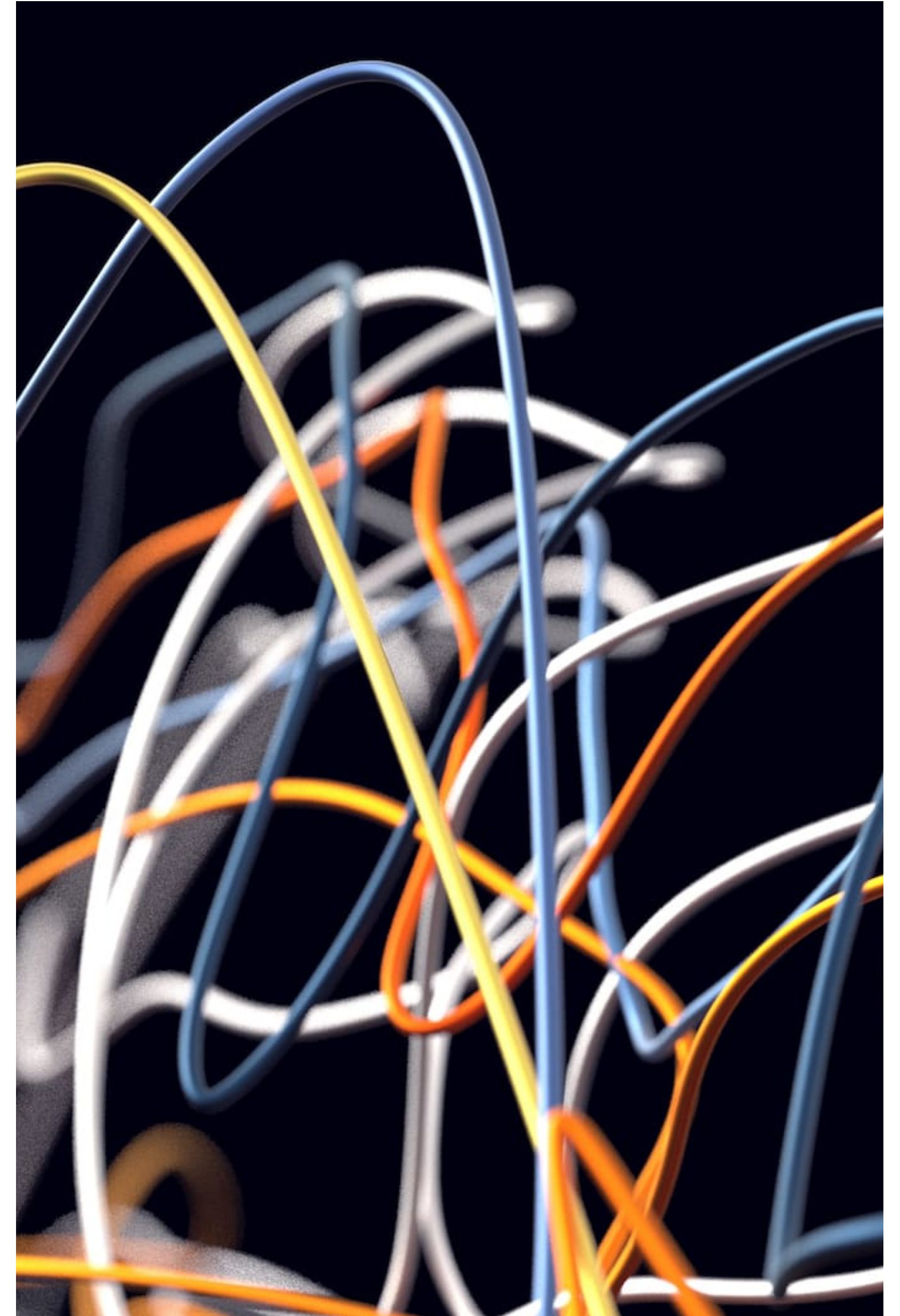
- Kein DOM
 - > nicht für statische Websites
 - > Landing Page mit Button, der weiterleitet
- Große Bundle Size
- Standardmäßig PWA
- Gut für "richtige" Programme (Excel, Miro, Figma, ...)



Flutter & Desktop

Flutter Desktop

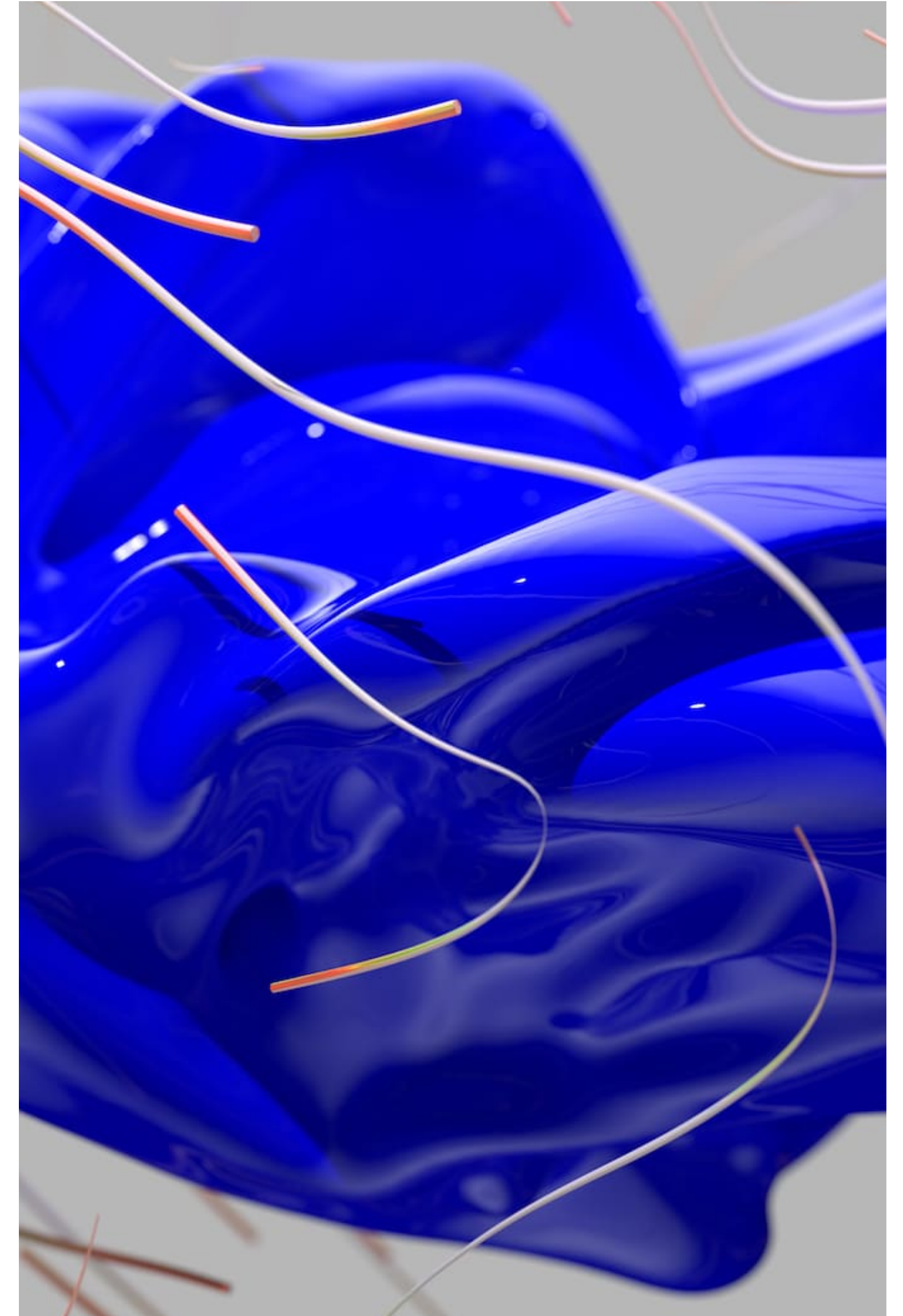
- Super Performance
- Leicht Desktop-App ins Web zu bringen
- fluent_ui & macos_ui für Desktop Aussehen
- Wenn Flutter Kompetenzen vorhanden, ernsthafte Alternative zu Web-Apps



Flutter & Embedded

Flutter Embedded

- Toyota baut Infotainment System mit Flutter
- Google Nest Hub mit Flutter
- Wenig Erfahrungsberichte



Probiert Flutter mal aus!



Torben Keller
torben.keller@innoq.com
linkedin.com/in/torben-keller/

innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim
+49 2173 3366-0

Ohlauer Str. 43
10999 Berlin

Ludwigstr. 180E
63067 Offenbach

Kreuzstr. 16
80331 München

Hermannstrasse 13
20095 Hamburg

Erftstr. 15-17
50672 Köln

Königstorgraben 11
90402 Nürnberg