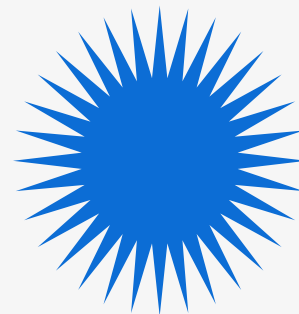


# QUICK WINS

## HOW TO MAKE YOUR APP MORE ACCESSIBLE



# SIMON AUER



**marqably**

from Austria

Co-Organizer FlutterVienna



<https://x.com/SimonEritsch>

<https://www.linkedin.com/in/simon-auer/>

[simon.auer@marqably.com](mailto:simon.auer@marqably.com)



---

## THIS TALK

<https://x.com/SimonEritsch>

<https://www.linkedin.com/in/simon-auer/>

---

# INTRO

## ACCESSIBLE APPS

- ★ Are apps that everyone can use
- ★ Follow guidelines like WCAG 2.2 (Web Content Accessibility Guidelines)
  - Level A - Basic Accessibility
  - Level AA - Strong Accessibility
  - Level AAA - Excellent Accessibility

# MOST COMMON BARRIERS



## Motor

Limited mobility, Dexterity issues, ...



## Visual

Blindness, Low Vision, Color Blindness, ...



## Hearing

Deafness, Hard of Hearing, ...



## Cognitive & Learning

Memory, Attention, Dyslexia, ...

---

# NUMBERS

✦ **15%**  
**of global population**  
live with a disability  
(acc. to WHO)

✦ **93%**  
**of websites & apps**  
are inaccessible or partly  
inaccessible

✦ **5-8h**  
**work per app**  
to bring your app to  
WCAG 2.2 Level A

✦ **50%**  
**more accessible**  
to reach or help people  
using your app

---



# Motor

Limited mobility, Dexterity issues, ...

---

## MIN ACTION ELEMENT SIZE

- Make sure interactive elements are at least 48 x 48 dp
- Avoid complex gestures or offer alternatives
- Provide visual feedback on taps, ...
- ...





# Visual

Blindness, Low Vision, Color Blindness, ...

---



**CONSIDER COLOR CONTRAST**

---

## POOR COLOR CONTRAST

- Too little contrast front/background color
- Especially for people with color blindness



I LOVE  
CHEESE



I LOVE  
CHEESE

## SOLUTION

- Make sure elements have enough contrast
  - **4.5 to 1:** text less than 24px (18px bold)
  - **3 to 1:** text at least 18 pt (14 pt bold)
- Use highContrast MediaQuery

main.dart

```
MediaQuery.of(context).highContrast ? highContrastBgColor : BgColor
```



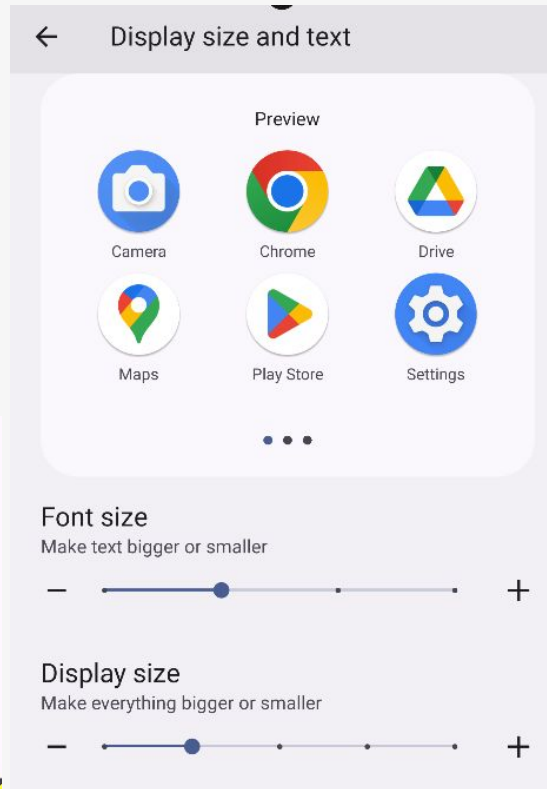
**MAKE FONT SIZE SCALABLE**

---

# MAKE FONT SIZE SCALABLE

- OS Settings exist for Font/Element scaling
- Text scales automatically in Flutter, but might break with layout
- Icons do not scale automatically

Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bookmarksgrove right at the coast of the Semantics, a large language ocean. A small river named Duden flows by their place and supplies it with the necessary regalia. It is a paradisematic country, in which roasted parts of sentences fly into your mouth. Even the all-powerful Pointing has no control about the blind texts it is an almost unorthographic life One day however a small line of blind text by the name of Lorem Ipsum decided to leave for the far World of Grammar. The Big Oxmox advised her not to do so, because there were thousands of bad Commas, wild Question Marks and devious Semikoli,



## SOLUTION

- Test your layouts with the highest scaled text/element settings
- Use **textScalerOf** to resize icons with text

highContrastMediaQuery.dart

```
1 final textScaleFactor = MediaQuery.textScalerOf(context);  
2  
3 return Icon(  
4   icon,  
5   size: textScaleFactor.scale(fontSize) * 1.2,  
6 );
```



**PREPARE FOR SCREEN READERS**

---



# PREPARE FOR SCREEN READERS

- Buttons with Icon and no explanation
- = > **Semantics()** Widget
- Image with important data and no explanation
- = > **semanticLabel Property**
- Layout element that is read out loud
- = > **ExcludeSemantics()** Widget
- Elements, that belong together, but are read out separately
- = > **Merge Semantics()** Widget



# SEMANTICS() WIDGET

highContrastMediaQuery.dart

```
1 Semantics(  
2   label: 'Menu toggle button',  
3   child: GestureDetector(  
4     onTap: () { ... },  
5     child: const Icon(Icons.menu),  
6   ),  
7 ),
```

# SEMANTIC PROPS

semantics.dart

```
1 Semantics(  
2   label: 'Go to previous chart',  
3   // this will announce the label when the button is updated  
4   liveRegion: true,  
5   // flip up gesture  
6   onIncrease: () { __performIncrease(); },  
7   // flip down gesture  
8   onDecrease: () { __performDecrease(); },  
9  
10  child: const Icon(Icons.arrow_left),  
11 )
```

# IMAGE SEMANTICS

highContrastMediaQuery.dart

```
1 Image.asset(  
2   'assets/chart_image.png',  
3   semanticLabel: 'A bar chart showing my sales', // ← label property  
4 ),
```

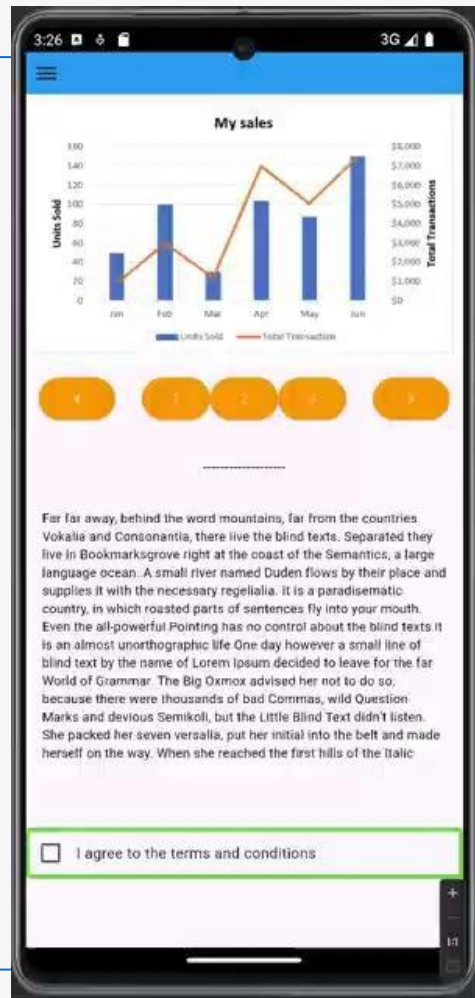
# EXCLUDE SEMANTICS

semantics.dart

```
1 ExcludeSemantics(  
2   child: Text('-----')  
3 ),
```

# PREPARED FOR SCREEN READERS

- Added semantics widgets around buttons/icons, ...
- Added semantic label to button
- Excluded the dash visual divider from semantics
- Merged the checkbox and checkbox label to one element





# **EXPERIENCE TIPS**

---

# ENFORCE A11Y WITH WIDGETS

- Create widgets, that enforce a11y
- Buttons, Images, ...

semantics.dart

```
1 class Btn extends StatefulWidget {  
2   const Btn({  
3     required this.semanticLabel,  
4     required this.onTap,  
5     ...  
6   });  
7  
8   ...  
9 }
```



# WRITE A11Y TESTS

semantics.dart

```
1 testWidgets('implemented a11y criteria and guidelines', (tester) async
  {
2     final handle = tester.ensureSemantics();
3     await tester.pumpApp(MyAwesomeButton(dog: dog));
4
5     // text contrast ratio is at least 4.5:1
6     await expectLater(tester, meetsGuideline(textContrastGuideline));
7
8     // ios only check for tappable size
9     await expectLater(tester, meetsGuideline(iOSTapTargetGuideline));
10
11    // android only check for tappable size
12    await expectLater(tester,
13      meetsGuideline(androidTapTargetGuideline));
14    ...
15
16    handle.dispose();
17  });
```

# USE A11Y DEBUGGER

- [https://pub.dev/packages/accessibility\\_tools](https://pub.dev/packages/accessibility_tools)  
by Rebel App Studio



Tap area is missing a semantic label

Tap area of 40x40 is too small:  
should be at least 44x44

×

UI settings

Reset all

Text scale: 1.0 ⓘ 

Reset

Localization ⓘ

Systemfi\_FIen\_US

Text direction ⓘ

SystemRTL LTR

Platform ⓘ

SystemandroidfuchsiaiOSlinuxmacOSwindows

Density ⓘ

Systemstandardcomfortablecompact

Bold text ⓘ

Systemonoff

Color mode simulation ⓘ

Offprotanopia deuteranopia tritanopia invertedgrayscale

Screen reader mode ⓘ

×

Close

# ENABLE FOR WEB

main.dart

```
1 import 'package:flutter/semantics.dart';  
2  
3 void main() {  
4   runApp(const MyApp());  
5   if (kIsWeb) {  
6     SemanticsBinding.instance.ensureSemantics();  
7   }  
8 }
```

## MORE INFORMATION

---

- ✳ [https://pub.dev/packages/accessibility\\_tools](https://pub.dev/packages/accessibility_tools)
  - ✳ **Vita Petrovskaya: Mobile App Accessibility: How to Make an App Inclusive?**  
<https://blog.flutter.wtf/mobile-app-accessibility/>
  - ✳ **Shawn Blais: Crafting a great experience for screen readers**  
<https://blog.gskinner.com/archives/2022/09/flutter-crafting-a-great-experience-for-screen-readers.html>
  - ✳ **VeryGoodVentures: Exploring Accessibility and Digital Inclusion with Flutter**  
<https://verygood.ventures/blog/exploring-accessibility-and-digital-inclusion-with-flutter>
  - ✳ **Thomas Steiner: Accessibility in Flutter on the Web**  
<https://medium.com/flutter/accessibility-in-flutter-on-the-web-51bfc558b7d3>
  - ✳ **Muhammed Salih Guler: A deep dive into Flutter's accessibility widgets**  
<https://medium.com/flutter-community/a-deep-dive-into-flutters-accessibility-widgets-eb0ef9455bc>
-

# THANK YOU <3



**marqably**

from Austria

Co-Organizer FlutterVienna



<https://x.com/SimonEritsch>

<https://www.linkedin.com/in/simon-auer/>

[simon.auer@marqably.com](mailto:simon.auer@marqably.com)

---