Fluttercon Europe 2024

Testing that your app is accessible to all

Alexander Troshkov

REBEL APP STUDIO

BY CODEMATE

Accessibility misconceptions

“It’s a separate feature that we’ll add at some point in the future”

“Only 0.1% of our users will use it”

“It’s hard to implement, test and maintain”

“Nothing’s gonna happen if we don’t add it”

The European Accessibility Act (Directive 2019/882) is a landmark EU law which requires some everyday products and services to be accessible for persons with disabilities.

...

As of 28 June 2025, customers will be able to file complaints before national courts or authorities if services or products do not respect the new rules.

ec.europa.eu

Accessibility testing

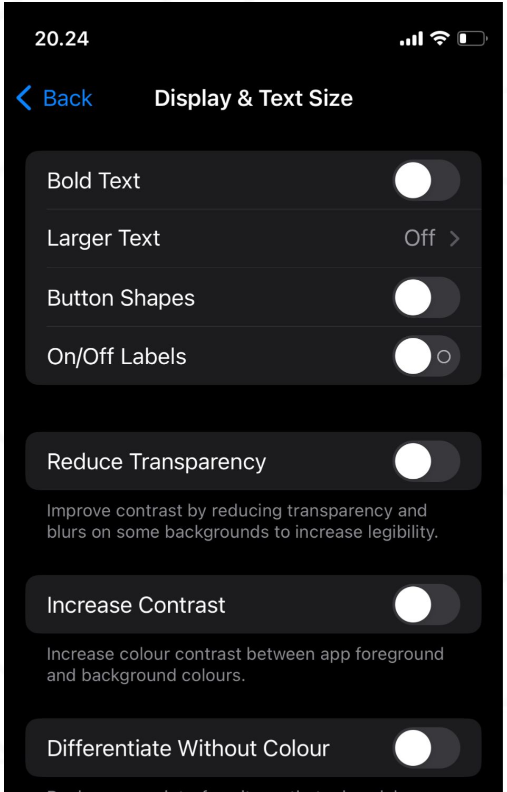
WHOOSH
AND YOUR APP IS ACCESSIBLE

1. No testing

2. Manual testing

Manual testing

iOS text settings



20.24

Back

Display & Text Size

Bold Text ○

Larger Text Off >

Button Shapes ○

On/Off Labels ○

Reduce Transparency ○

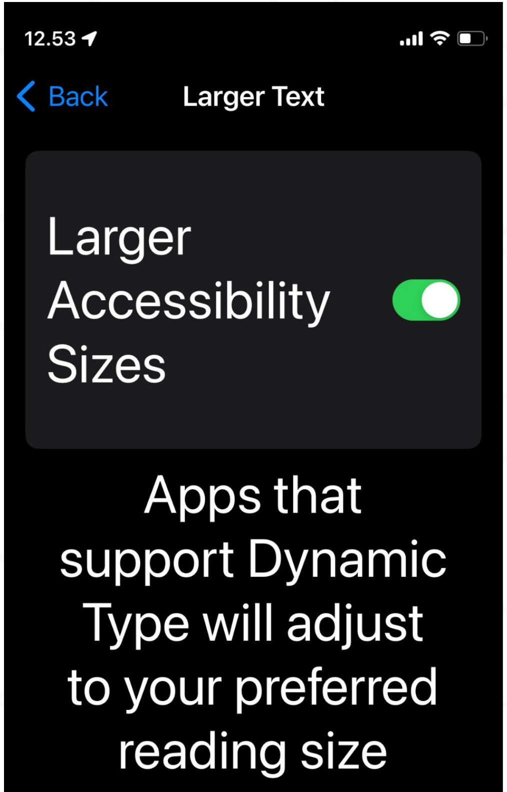
Improve contrast by reducing transparency and blurs on some backgrounds to increase legibility.

Increase Contrast ○

Increase colour contrast between app foreground and background colours.

Differentiate Without Colour ○

Setting text size to huge



12.53

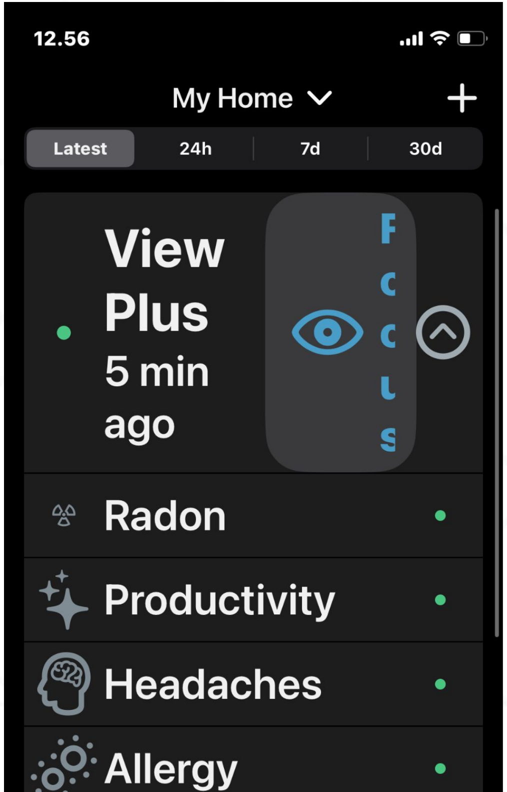
Back

Larger Text

Larger Accessibility Sizes ●

Apps that support Dynamic Type will adjust to your preferred reading size

Some apps break



12.56

My Home +

Latest 24h 7d 30d

View Plus 5 min ago

Focus

Radon

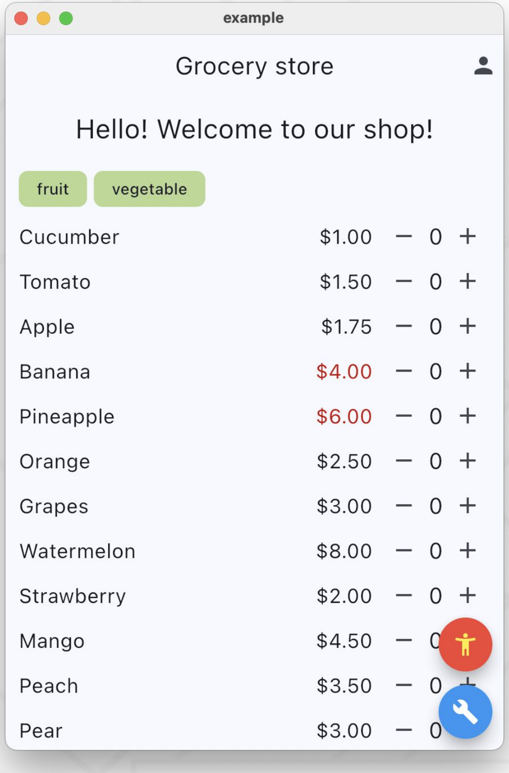
Productivity

Headaches

Allergy

2a. accessibility_tools

Accessibility tools



example

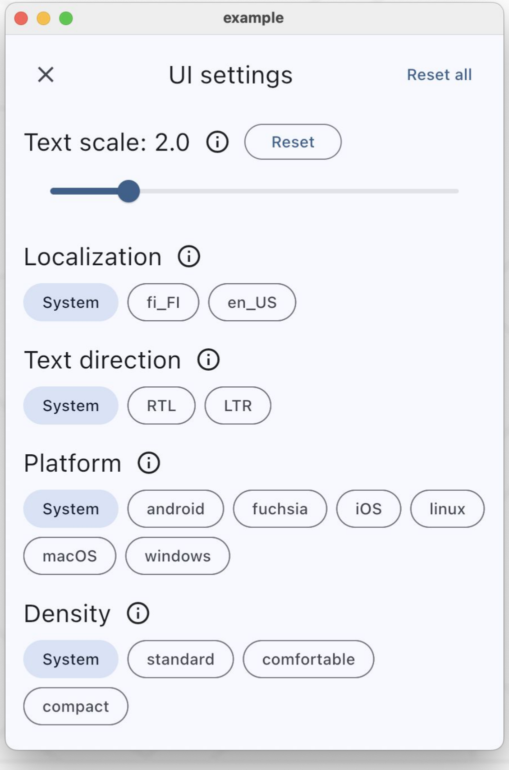
Grocery store

Hello! Welcome to our shop!

fruit vegetable

Cucumber	$1.00	−	0	+
Tomato	$1.50	−	0	+
Apple	$1.75	−	0	+
Banana	$4.00	−	0	+
Pineapple	$6.00	−	0	+
Orange	$2.50	−	0	+
Grapes	$3.00	−	0	+
Watermelon	$8.00	−	0	+
Strawberry	$2.00	−	0	+
Mango	$4.50	−	0	
Peach	$3.50	−	0	+
Pear	$3.00	−	0	

My app



example

× UI settings Reset all

Text scale: 2.0 ⓘ Reset

Localization ⓘ

[✓ System] fi_FI en_US

Text direction ⓘ

[✓ System] RTL LTR

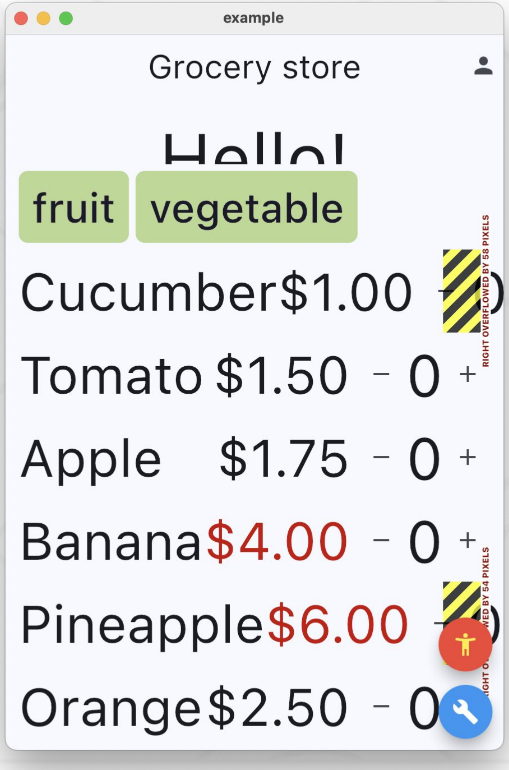
Platform ⓘ

[✓ System] android fuchsia iOS linux macOS windows

Density ⓘ

[✓ System] standard comfortable compact

A11Y tools testing panel



example

Grocery store

Hello!

fruit vegetable

Cucumber$1.00

RIGHT OVERFLOWED BY 58 PIXELS

Tomato $1.50 − 0 +

Apple $1.75 − 0 +

Banana$4.00 − 0 +

Pineapple$6.00

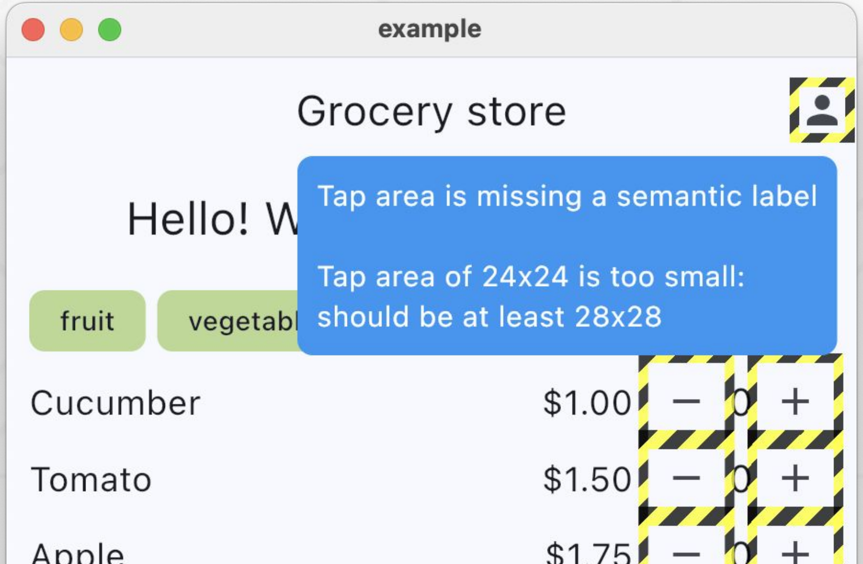
WED BY 54 PIXELS

Orange$2.50 − 0

My app is broken :(

Accessibility tools

Accessibility tools found missing semantic labels



Tap area is missing a semantic label

Tap area of 24x24 is too small: should be at least 28x28

3. Automated testing

Flutter accessibility guidelines

- Tap target size
- Tap target label
- Text contrast
- Option to create your own guidelines

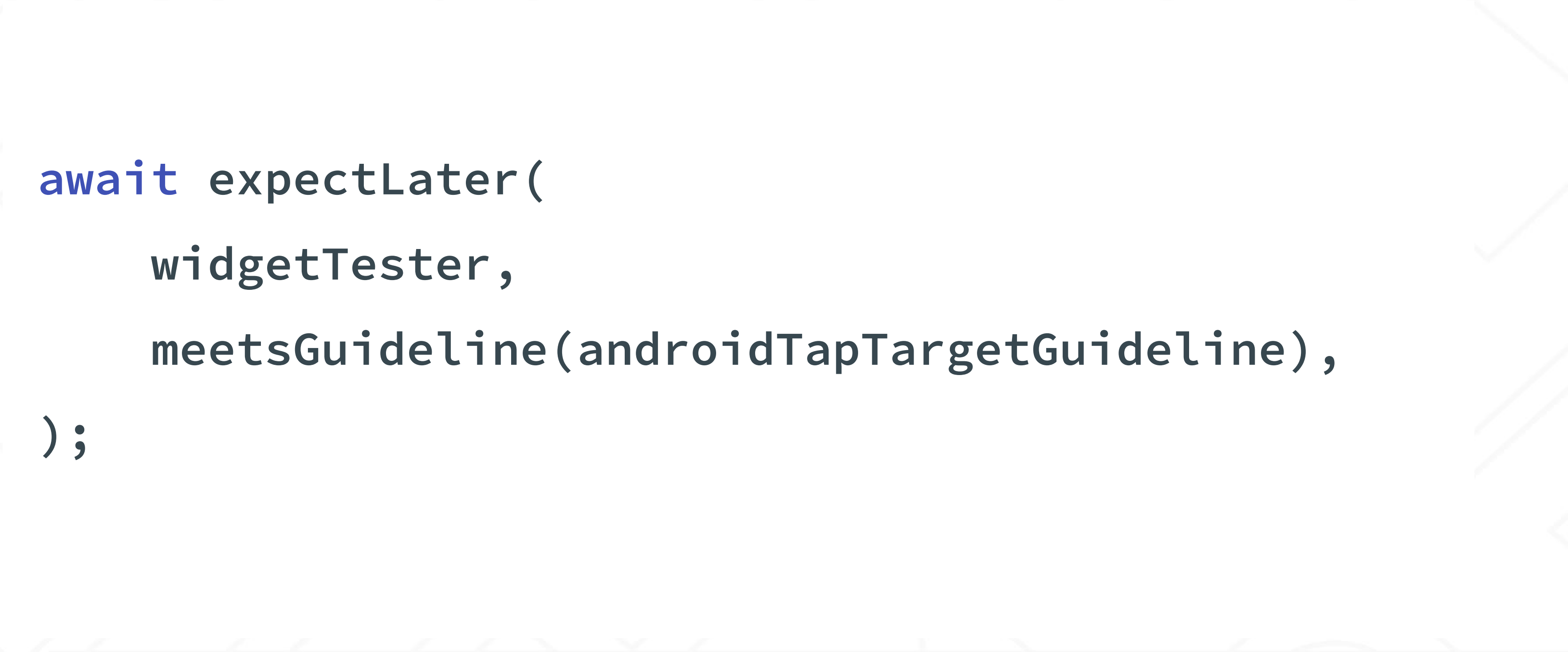
Using accessibility guidelines

```
final SemanticsHandle handle = widgetTester.ensureSemantics();
```

Using accessibility guidelines

```dart
final SemanticsHandle handle = widgetTester.ensureSemantics();

// 1. Your widget test code here

// 2. Guideline stuff

// 3. The rest of your widget test code here
```

Using accessibility guidelines



```
await expectLater(
    widgetTester,
    meetsGuideline(androidTapTargetGuideline),
);
```

Using accessibility guidelines

```
handle.dispose();
```

Creating your own guideline

Creating your own guideline

```
class LabelLengthGuideline extends AccessibilityGuideline {
```

Creating your own guideline

```dart
@override
String get description => 'Check that semantic labels are not too long.';
```

Creating your own guideline

```
@override
FutureOr<Evaluation> evaluate(WidgetTester tester) {
```

Creating your own guideline

```
Evaluation result = const Evaluation.pass();
```

Creating your own guideline

```
for (final RenderView view in tester.binding.renderViews) {
  result += _traverse(view.owner.semanticsOwner.rootSemanticsNode!);
}
```

Creating your own guideline

```
Evaluation _traverse(SemanticsNode node) {
    Evaluation result = const Evaluation.pass();

    // TODO: check nodes here

    return result;
}
```

Creating your own guideline

```
node.visitChildren((SemanticsNode child) {
  result += _traverse(child);
  return true;
});
```

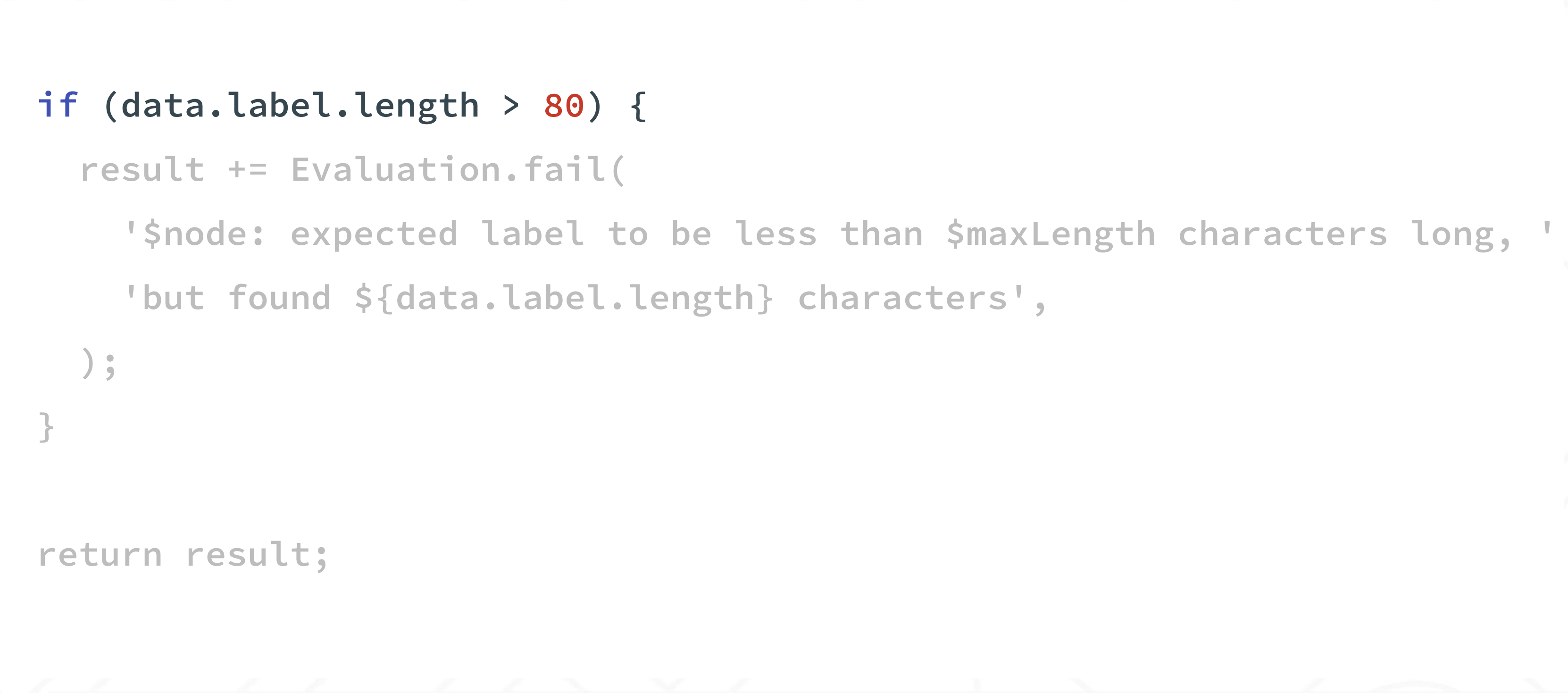
Creating your own guideline

```
if (node.isMergedIntoParent ||
    node.isInvisible ||
    node.hasFlag(SemanticsFlag.isHidden)) {
    return result;
}
```

Creating your own guideline

```
final SemanticsData data = node.getSemanticsData();
```

Creating your own guideline



```
if (data.label.length > 80) {
  result += Evaluation.fail(
    '$node: expected label to be less than $maxLength characters long, '
    'but found ${data.label.length} characters',
  );
}

return result;
```

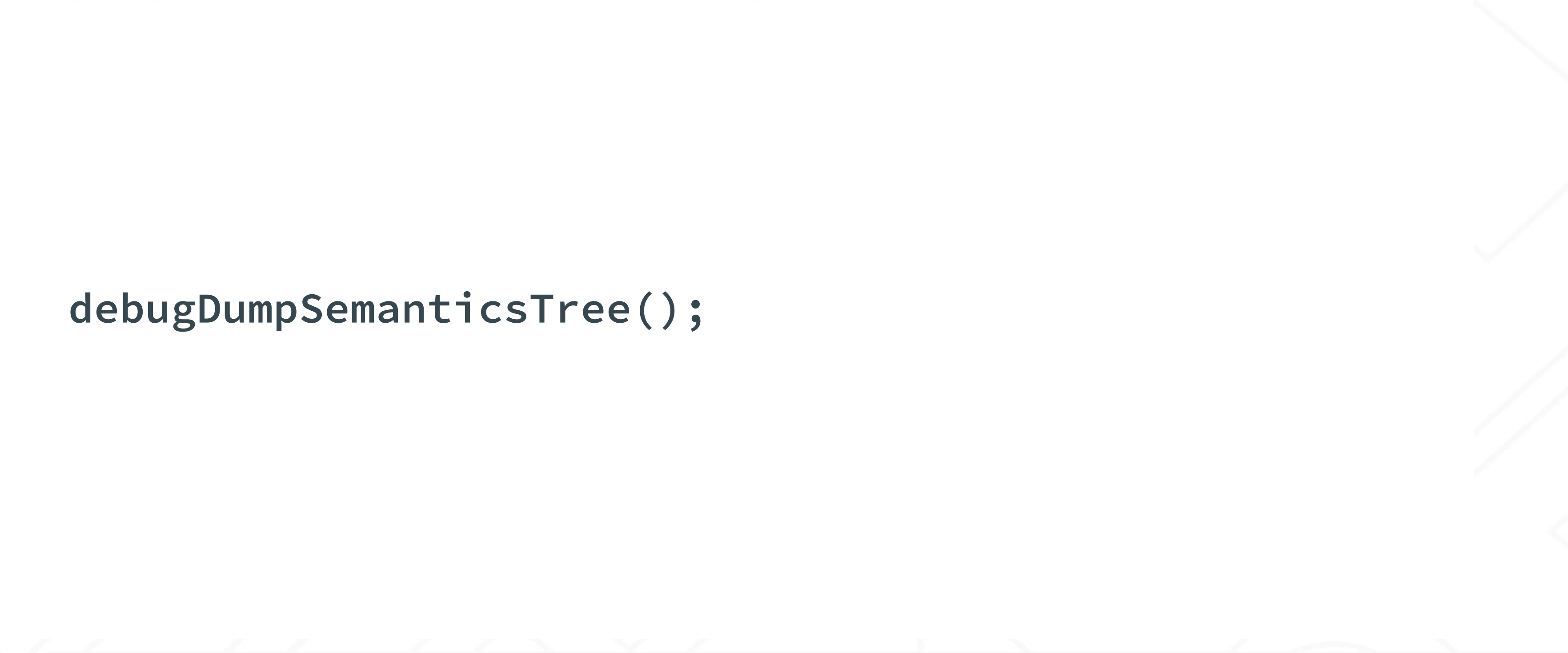
Creating your own guideline

```
if (data.label.length > 80) {
  result += Evaluation.fail(
    '$node: expected label to be less than $maxLength characters long, '
    'but found ${data.label.length} characters',
  );
}

return result;
```

Creating your own guideline

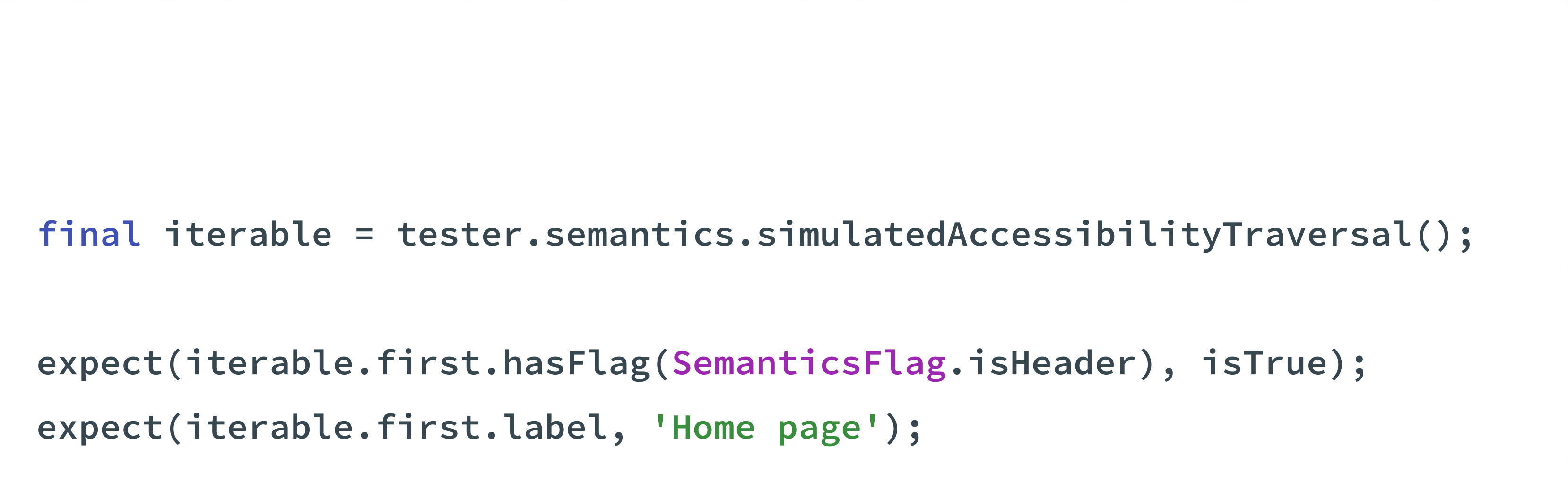
```
await expectLater(
    widgetTester,
    meetsGuideline(LabelLengthGuideline()),
);
```

Note: semantics in tests



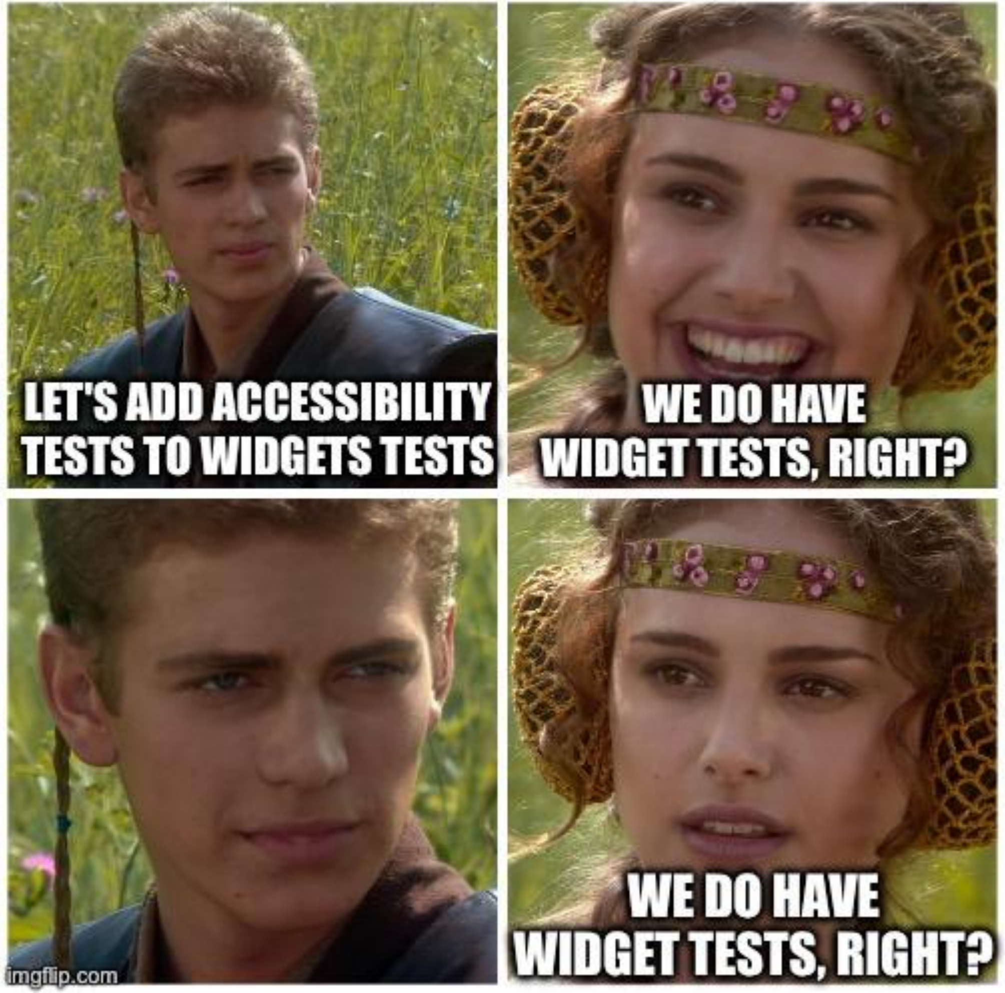
```
debugDumpSemanticsTree();
```

Note: semantics in tests



```
final iterable = tester.semantics.simulatedAccessibilityTraversal();

expect(iterable.first.hasFlag(SemanticsFlag.isHeader), isTrue);
expect(iterable.first.label, 'Home page');
```

Running existing tests with different accessibility settings


LET'S ADD ACCESSIBILITY TESTS TO WIDGETS TESTS
WE DO HAVE WIDGET TESTS, RIGHT?
WE DO HAVE WIDGET TESTS, RIGHT?
imgflip.com

Accessibility + existing tests

- `debugSemanticsDisableAnimations`
- `TextScaler`
- `boldText`
- `highContrast`

Accessibility + existing tests

```
flutter test test/mytest.dart
           --dart-define=testFlavor=a11yTester
```

Accessibility + existing tests

```
const testFlavor = String.fromEnvironment('testFlavor');
const extremeConditions = testFlavor == 'a11yTester';
```

Accessibility + existing tests

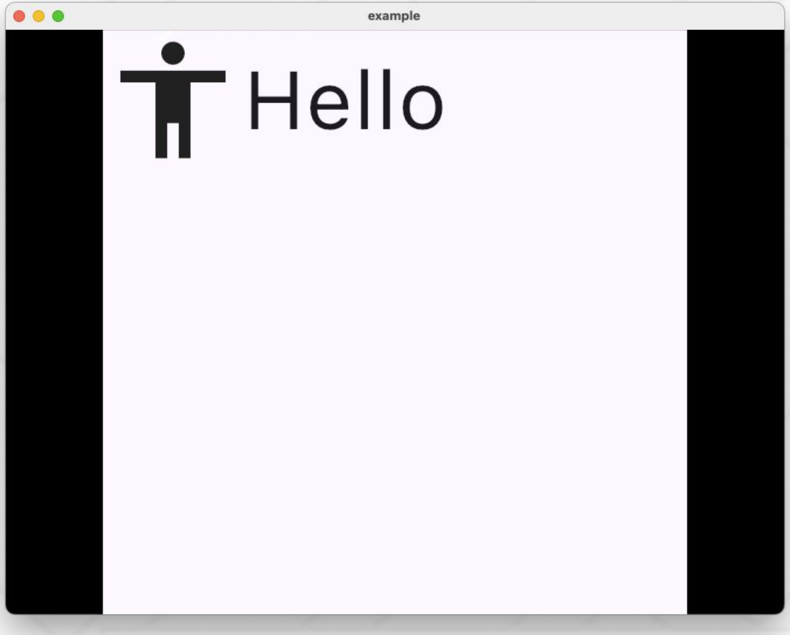
```
double textScaler = 1.0;
bool boldText = false;
bool highContrast = false;

if (extremeConditions) {
    debugSemanticsDisableAnimations = true;
    textScaler = 3.0;
    boldText = true;
    highContrast = true;
}
```

Accessibility + existing tests

```
await tester.pumpWidget(
    MaterialApp(
        home: Scaffold(
        body: MediaQuery(
            data: MediaQueryData(
                textScaler: TextScaler.linear(textScaler),
                boldText: boldText,
                highContrast: highContrast,
            ),
            child:
```

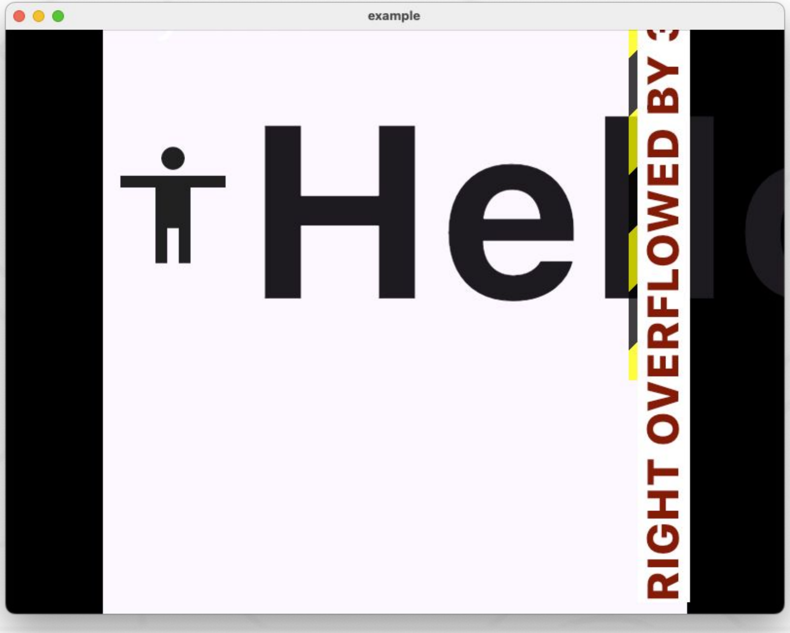
Accessibility + existing tests

Default settings: test passes



Hello

Accessibility options tweaked:
Text overflows and test fails



Hello

RIGHT OVERFLOWED BY

TODO: encourage people to check their app’s accessibility

Testing that your app is accessible to all

Alexander Troshkov

github.com/aednlaxer

Slides and materials: github.com/aednlaxer/flutterconeurope2024

Accessibility tools: pub.dev/packages/accessibility_tools

Come talk to us!

We're hiring!
TEST YOUR MATCH
Test your match with Rebel App Studio by answering a few simple questions!
REBEL APP STUDIO
GAMING & LOUNGE
SPEAKER
GLASS

REBEL APP STUDIO BY CODEMATE