



UNIVERSITY OF
PADUA

Mobile content accessibility guidelines on the Flutter framework

Master Degree in Computer Science
Matteo Budai
18 July 2024

ID Number - 2057217

Supervisor - Prof Ombretta Gaggi



DIPARTIMENTO
MATEMATICA

Accessibility

Accessibility = Possibility of accessing information and services available online by diversified categories of users and from a range of different devices



Cognitive &
Learning
Disabilities



Blindness
Low Vision
Color-blindness



Speech Inputs



Hearing
Impairment



Motor &
Dexterity

Disadvantages classes:

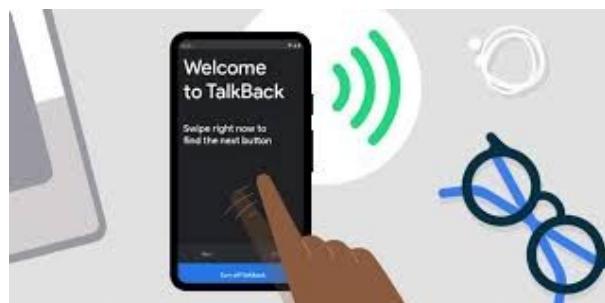
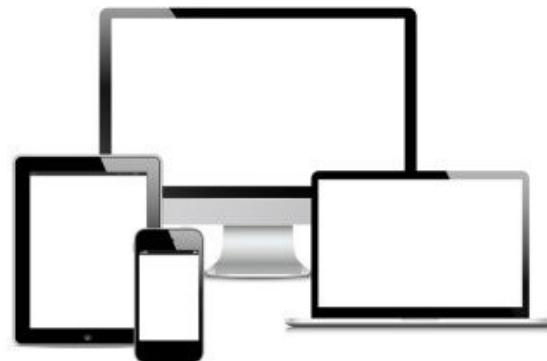
- Sight
- Movement
- Hearing
- Convulsions
- Intellect

Mobile issues and Legislation



Mobile issues:

- Touchscreen interfaces
- Small screen sizes
- Different input modalities
- Device use in different settings ...

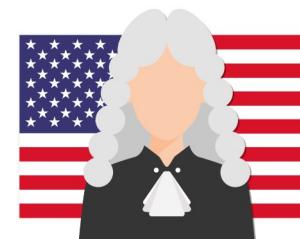


Useful Events:

- Apple's VoiceOver
- Google TalkBack screen reader

Legislation:

- ❖ Americans with Disabilities
- ❖ (Usa) Section 508 of Rehabilitation Act
- ❖ (ITA) Legge Stanca
- ❖ AGID - 28 June 2025 not possible to place digital programs not accessible



W3C = World Wide Web Consortium



WCAG = Web Content Accessibility Guidelines

With W3C the goal is a *single shared standard* for web content accessibility

Focusing on people with disabilities

Evolution: - WCAG 1.0 (1999) - 14 principles

- WCAG 2.0 (2008) - 12 guidelines

- WCAG for mobile (2015)

- WCAG 2.1 (2018) - +1 guideline; +17 success criteria

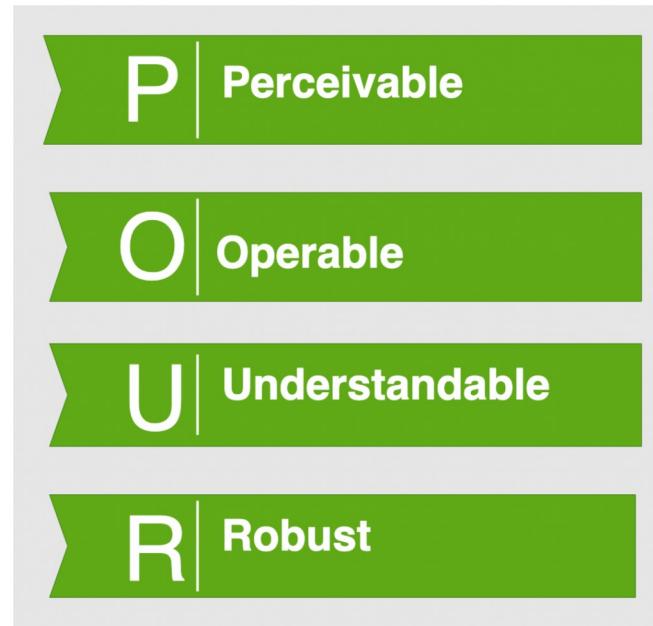
- WCAG 2.2 (2023) - +9 success criteria and 1 removed



Organization and Principles

WCAG 2.2 - 13 guidelines and 86 success criteria

Organised under four principles:



User interfaces components and information must be delivered to users in ways they can perceive

Users must be able to operate the interface

Users must be able to understand the information as well as the operation of the user interface

Users must be able to access the content as technologies advance

Each success criteria:



'A' → Conformance Priority 1

'AA' → Conformance Priority 1 and 2

'AAA' → Conformance Priority 1, 2 and 3

MCAG = Mobile Content Accessibility Guidelines

- An official document provided by W3C (2015)
- *Principles* (same of WCAG) and *guidelines* applied to mobile context
- Built for WCAG 2.0 VALID → WCAG 2.2
- In the thesis *guidelines* adapted considering **new** success criteria of WCAG 2.1 and WCAG 2.2
- Definition of **new** ideas for **MCAG**



PERCEIVABLE

- ❖ *Typography*: Using accessible fonts or the font of the system
- ❖ *Themes*: Support both light and dark modes

OPERABLE

- ❖ *Voice Control*: Ability to fully control the application using advanced voice commands
- ❖ *Customization and simplification of the interface*
- ❖ *Action confirmations*: Ensure clear and understandable confirmation

UNDERSTANDABLE

- ❖ *Action progress display*: Clear and progressive indicators to understand the status
- ❖ *Avoid non necessary elements*
- ❖ *Destination of the navigation buttons*: Clearly distinguishable
- ❖ *Guided start-up tour*: Guide user through main features and options available

ROBUST

- ❖ *Advanced feedback*: More sophisticated and customizable feedback (facilitation)
- ❖ *Multilingual support*: Application translated for all languages it makes available
- ❖ *Provide users enough time to read and use the content*: Need to define timescales



FLUTTER

- Open source project conceived by Google in 2018
- Last version 3.0 (2023)
- Dart language
- Compiles to native offering high performance
- Easy to learn and performant
- Free
- Primarily used for Android and iOS applications
- Composed of 3 layers

DART

- Developed by Google in 2011
- Aimed at replacing JavaScript
- Type-safe
- Two platforms: Dart Native and Dart Web



Application

RESEARCH QUESTIONS

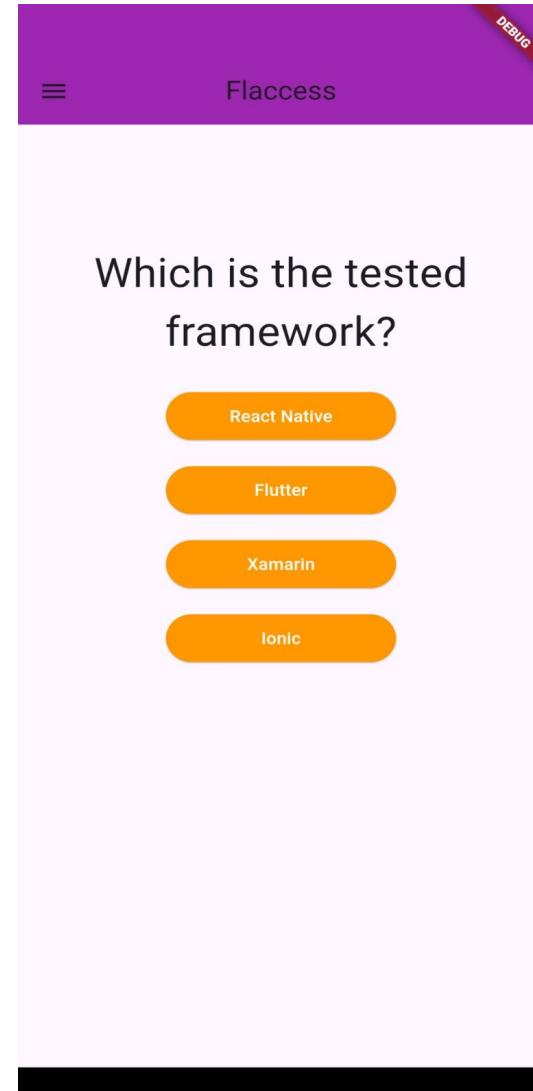
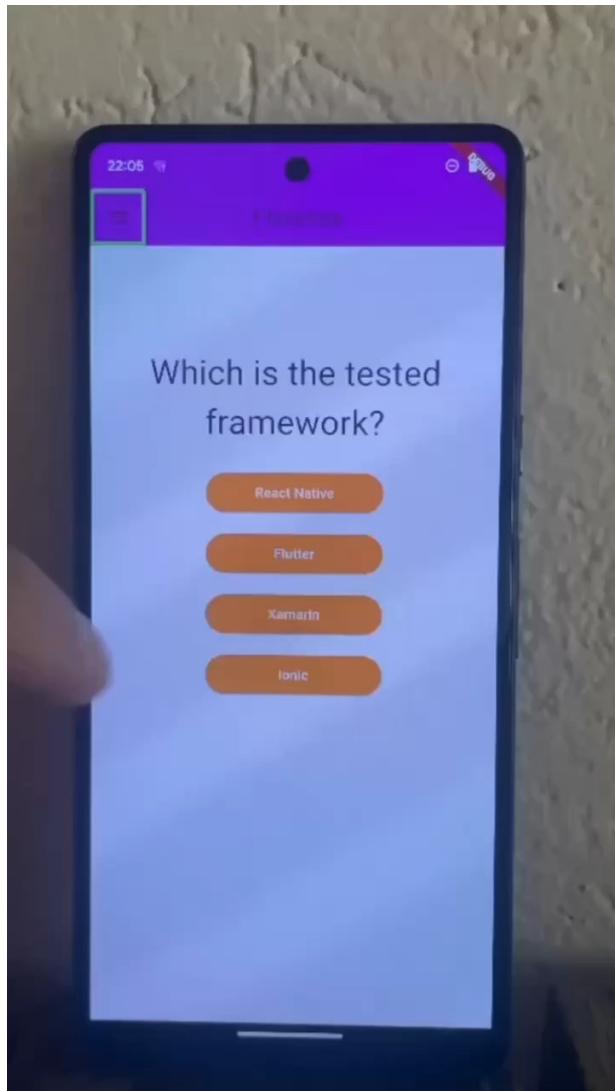
- RQ1: Are the elements and widgets provided by the framework accessible by default?
- RQ2: If an element or a widget is not accessible by default, is it possible to make it accessible?
- RQ3: If an element or a widget is not accessible by default, and can be made accessible by developers, how much does it cost in terms of additional required code?

TEST

- Creation of the Application called '*Flaccess*'
- Integrated Development Environment: *Android Studio*
- Devices: *Emulator* and *Android Google Pixel 7*
- Screen reader: *TalkBack*



Implementation 1



Which is the tested framework?

React Native

Flutter

Xamarin

Ionic

Implementation 1 Analysis



Widget/Component	Question 1	Question 2	Question 3
AppBar	✗	+1W +1P	2
Button (all)	✓		
IconButton	✗	+1P	1

Legend: ✓ = Accessible, ✗ = Not Accessible, +1W = Number of Widgets, +1P = Number of Properties, Question 3= Additional lines of code

Widget/Component	Question 1	Question 2	Question 3
Drawer	✓		
DrawerHeader	✗	+1W +1P	2
ListTile	✗	+1W +1P	2

Implementation 2



DEBUG

Documentation

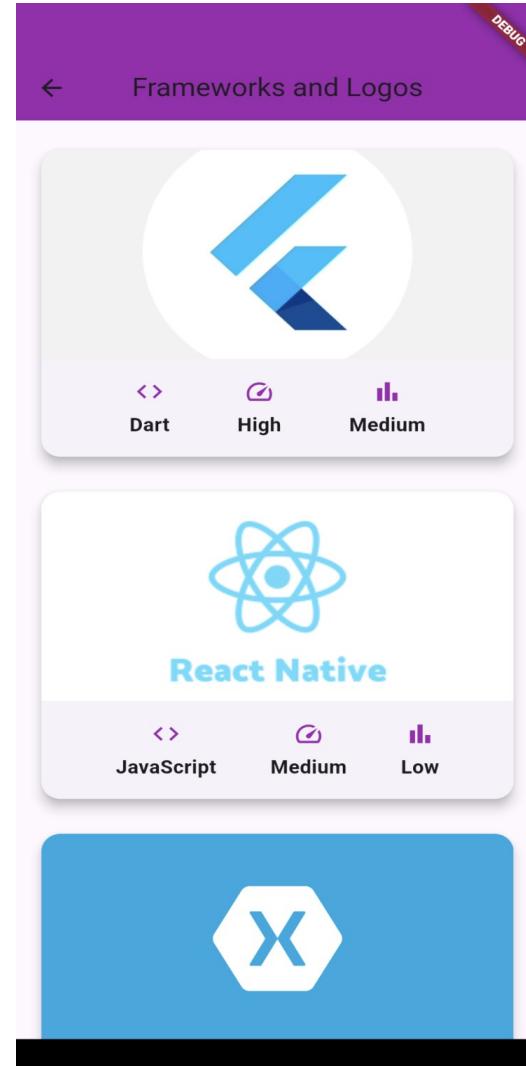
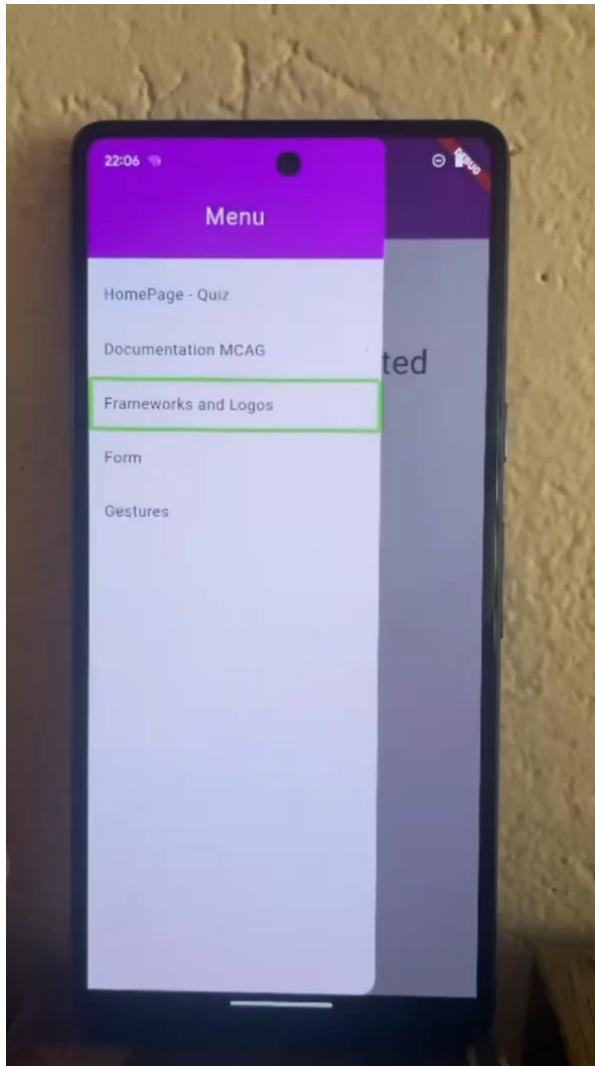
[Official WCAG guidelines for the mobile](#)

WCAG for mobile denominated as " Mobile Accessibility: HowWCAG 2.0 and Other W3C/WAI Guidelines Apply to Mobile" was an official document provided by W3C. It aims to pursue the accessibility of the content in the mobile world. Its principles, guidelines and success criteria can be applied to mobile web content, mobile web apps, native apps and hybrid apps using web components inside native apps. The document was published on 26th february 2015 providing recommendations, notes and guidelines of mobile technologies including mobile web best practices and mobile web application best practices. With the term "mobile" we can consider small handheld devices such as smartphones, wearables, for instance smart-watches and also large tablets. We have to note that the last version of WCAG, WCAG 2.2, can be applied and was relevant to both web and non-web mobile content and applications. But we know that mobile devices present a mix of accessibility issues that are different from desktop. WCAG 2.2 did not provide testable success criteria for some of the mobile-related issues

ANALYSIS

Widget/Component	Question 1	Question 2	Question 3
Text (link)	✓		
ScrollBar	✓		
SinglechildScrollView	✓		

Implementation 3



Implementation 3 Analysis



Widget/Component	Question 1	Question 2	Question 3
Image	✗	+1P	1
Card	✓		
Icon	✗	+1P	1
Text (header)	✗	+1W +1P	2
ClipRect	✓		
CircleAvatar	✓		

Implementation 4



The image shows two screenshots of a Flutter mobile application interface. Both screenshots have a purple header bar with a back arrow and the word "Form". A "DEBUG" watermark is visible in the top right corner of both.

Screenshot 1 (Left): This screenshot shows an empty form with the following fields:

- Name: (Text input field)
- Email: (Text input field)
- Password: (Text input field)
- Gender: Male (Dropdown menu showing "Male")
- Date of Birth: 2024-06-17 (Text input field) with a "Choose Date" button next to it.
- Accept Terms: (Switch control)
- Level of Satisfaction: (Slider control with a value of 8.0)
- Programming: (Text input field) with a dropdown menu below it containing:
 - Flutter
 - React Native
 - Xamarin
 - Ionic
- Submit: (Large blue button)

Screenshot 2 (Right): This screenshot shows the same form after the user has filled it out and submitted it. The data is displayed in a modal dialog box:

Form Submitted

Field	Value
Name	Matteo
Email	matteo@gmail.com
Password
Gender	Male
Date of Birth	1999-09-23
Accept Terms	Yes
Satisfaction	8.0
Ability	Programming
Framework	Flutter

At the bottom of the dialog is an "OK" button.

Implementation 4 Analysis



Widget/Component	Question 1	Question 2	Question 3
Form	✓		
TextField	✓		
DropdownButtonFormField	✓		
showDatePicker	✓		
SwitchListTile	✓		
RadioListTile	✓		
CheckBoxTile	✓		
Slider	✓		
AlertDialog	✓		

Implementation 5



A screenshot of a Flutter mobile application interface. At the top is a purple navigation bar with a back arrow and the text "Gestures". To the right of the text is a switch labeled "DEBUG". Below the navigation bar is a white area containing four colored squares stacked vertically: blue, red, green, and purple. In the bottom left corner of this area is a small circular button with the text "Reset Page".

ANALYSIS

Widget/Component	Question 1	Question 2	Question 3
GestureDetector	✓		

OTHER CONSIDERATIONS

Widget/Component	Question 1	Question 2	Question 3
Language	✗	+1W +1P	10
Layout	✓		

Conclusions

- *Mobile Content Accessibility Guidelines* need an **update** and an apposite **documentation**
- Most of the *Widgets* and *Elements* in *Flutter* are **accessible by default**
- Developers can use the thesis as **manual** for creating *Flutter* mobile applications **accessible to everyone**

