

# Mobile A11y testing

## Event Notes

Most of standard WCAG 2.x checkpoint success criterion still apply to mobile devices and mobile Apps.

Usability, ease of use, and full accessibility cannot really be tested by automation and require human users to perform mobile app accessibility testing. Mobile testing needs to happen with people using assistive technologies, and/or technologies that emulate assistive technologies under a mobile device format.

WCAG is the **best-practice** standard for Mobile accessibility testing.

**Advanced Input Systems:** Auto fill-out fields wherever possible including appropriate use of Autocomplete features.

**Readable and operable in portrait and landscape device orientation.** This means that the navigation and menu systems should flexibly adapt to the orientation of the screen.

**Consistent navigation and control positioning** – ensure that user nav and controls are consistently positioned, no matter what the screen orientation.

**Above the fold** - easy control access – Important controls and navigation items should be easy to access and positioned such that an e-reader user could easily activate such controls and links. Such elements should be above the visual fold of the page even when rotated. Positioning important page elements before the perceivable page-scroll-horizon assists users with visual and cognitive impairments. If the most important controls or elements on the page require scrolling, it can cause confusion and unnecessary document traversal.

**Controls and links** – sufficient space – to operate all controls and navigation links with fingers, there should be sufficient surrounding space such that the user can easily target and operate the controls using human fingers without difficulty, miss-targeting, or confusion.

**Touch-interaction points should be no smaller than 9mm** (0.35in) in both dimensions. (About 21-24px minimum typically @ 60DPI) but this depends on pixel-density on the device. Ideally, interactive elements should trigger on a mouse-Up action; when a mouse is used – to allow a user to move the cursor away from an accidentally clicked-down element.

**Indicators of available functionality should be evident.**

E.g. if a swipe-left feature is expected, an indicator of such should be present.

**Minimalism** -Mobile App data should typically be in an easy to read and minimalistic form for mobile devices. Space available is small and page real-estate is precious and should not be wasted with unnecessary content.

**Ads and moving content** – a common plague to Mobile devices is changing or appearing advertisements will appear and distract the reader with animations or flashes as well as shifting or moving content from where the reader was looking or entirely out of the viewport. **This is an egregious and wholly unacceptable practice. Don't do it!**

**Keyboard Usage** – it is a common misconception that keyboard-like devices are not used on mobile devices and Applications. There absolutely are systems that provide keyboard like functionality for mobile devices – such as offering swiping and typing as a proxy into the mobile app page-content. Don't ignore keyboard operability and therefore visual focus and appropriate tab-index functionality.

Remember that even regular keyboard and mouse devices can be operated on a mobile device – don't presume a no-keyboard situation.

**Liquid designs (flexible adapting CSS layouts)** – should invoke the appropriate CSS Breakpoints. That's to say that CSS @media declarations should be appropriate for common mobile devices.

**E.g.** @media only screen and (min-width: 844px) and (orientation: landscape) { styles go here... }

**Touch and Gesture** – some devices offer special touch-screen gestures and device manipulation such as shaking. These features should not be disturbed, and where useful, leveraged for good use within your mobile app.

**Don't disable device or platform/OS specific features** –text-selection and zooming, copy and pasting should function as designed by the device without interference. At the same time don't rely on them – for instance, some individuals may not be able to reliably “shake” the device and so such an action should not be relied upon, and alternative mechanisms should be made available.

**E.g.** Swipe, Pinch, Tap, Long press, twirl and tilt, shake, drag, sling.

**Set appropriate virtual keyboard** – some devices provide alternative keyboards (virtual keyboards) that can be extremely useful depending on the expected data. Set the best format where possible. E.g. If expecting numbers only try to set the virtual keyboard to numbers (enable virtual keypad) etc.

## Appendix

**Mobile Accessibility at W3C:**

<https://www.w3.org/WAI/standards-guidelines/mobile/>

**How WCAG 2.0 and Other W3C/WAI Guidelines Apply to Mobile:**  
<https://www.w3.org/TR/mobile-accessibility-mapping/>