Transportation App User Accessibility Testing Best Practices & Checklist

Created by Feonix – Mobility Rising and Menlo Innovations as part of the MI Ride Paratransit Project, sponsored by the Michigan Department of Transportation and the Regional Transit Authority of Southeast Michigan

Author Valerie Lefler, Founder & Executive Director, Feonix - Mobility Rising

Table of Contents

Introduction

Mobile application accessibility is essential to ensuring that as your services support riders with disabilities, so does the technology to request and book the trip itself. The process to establish and maintain transit technology accessibility for paratransit passengers involves many complicated, moving parts. Because of the complexity, this guide was created to support agencies and transportation technology vendors who are committed to ensuring accessibility for all riders.

In early 2019, the Michigan Department of Transportation awarded a grant to the Regional Transit Authority of Southeast Michigan to pilot an innovative technology solution that would allow users to manage and book paratransit rides across the region more accurately and expediently. Feonix - Mobility Rising was honored to take the role of project integrator to deploy the technology and coordinate the deployment of services with the riders and transit agencies.

As part of the project, we at Feonix had the opportunity to work hand in hand with Menlo Innovations and two different technology companies to take standard transit apps and mold them into accessible applications based on user experiences and testing. This document has been put together to share our learnings with additional stakeholders in the transportation industry.

Surprisingly, no standard process has been developed for software companies to follow in their quest for mobile app accessibility. The closest resource is the Web Content Accessibility Guidelines (WCAG), a comprehensive set of criteria for accessible digital content. Originally created for websites, today WCAG is regarded as the best-practice standard for all types of digital experiences, including mobile apps.

As no two people experience disability the same way, no two accessibility accommodations are quite the same. This complicates the creation of a linear checklist of recommendations as code and native app features evolve.

WCAG guidelines are based on four key principles known by the acronym, POUR: Perceivable, Operable, Understandable, and Robust. These principles guide the design of accessible technology.

- **Perceivable** - users can easily identify content and interface elements visually and by sound.
- **Operable** - users can use controls, buttons, and other interactive elements to access and operate the application not only by clicking, tapping, or swiping but also using keyboard or voice commands.
- **Understandable** - the information and user interface are consistent in presentation and format, and are predictable in design and usage patterns. Users should be able to comprehend the content and remember how to use the interface.
- **Robust** - content must be robust enough to be interpreted reliably by a wide variety of users and types of assistive technologies. As technologies evolve, code and content should remain accessible for users of common and current assistive devices and tools.

While WCAG itself is not a regulation or legislation, it is accepted as the global standard for web accessibility. In fact, legal compliance with many international regulations – including Section 508 of the Rehabilitation Act in the U.S – requires WCAG conformance.

WCAG does not include separate guidelines for making sure mobile apps and mobile versions of websites are accessible. However, transportation app developers and the agencies who fund them should look to WCAG as the closest thing to the gold standard for mobile app accessibility. It should also be noted that the apps must be designed to function properly across devices other than mobile phones – including tablets, smart TVs, and wearables – so that users can book rides and/or receive notifications.

Accessibility Checklist & Guidance for Best Practices in Transportation App Design & Development

Below and on the following pages you will find seven key areas to review and consider in making your transportation apps accessible. While the checklist below is extensive, additional work in testing the app is recommended using automated and current customer individual and group users – at deployment and before releasing major and minor updates and bug fixes. For example, a few small tweaks to the back end of the code can render the app unusable to screen readers.

Screen Layout

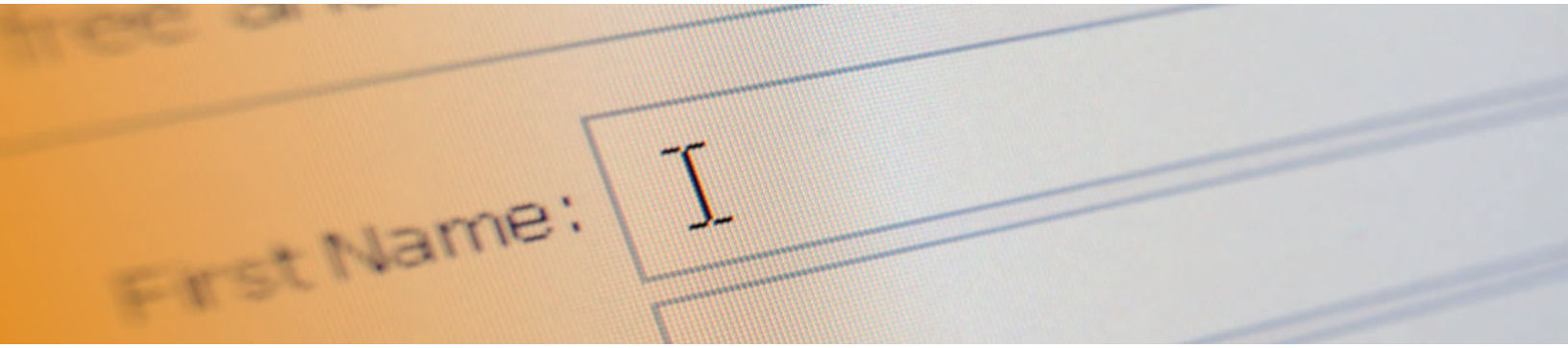
- [ ] Screen layout is consistent and predictable
- [ ] Pages have descriptive titles
- [ ] Headings are properly labeled, nested, in the correct reading order, and available to assistive technology
- [ ] Screen orientation can change between portrait and landscape without losing fidelity of the technology or input access

Menus & Navigation

- [ ] Menus, controls, and links work with touch and keyboard
- [ ] Menus, controls, and links are clearly labeled with and without assistive technology

Buttons & Controls

- [ ] Buttons and controls are large enough to view and select by touch, with the smallest interactive zone no less than 9 x 9 mm
- [ ] Buttons and controls are recognized properly by screen reader with clear and understandable descriptions
- [ ] Buttons and controls clearly visually indicate when they have been selected
- [ ] Buttons and controls have enough space around them so neighboring controls are not selected by mistake
- [ ] Buttons and controls are placed where they can be accessed easily, regardless of how the device is held. Developers should consider how an easy-to-use button placement for some users might cause trouble for others (for example, left- versus right-handed use or assuming the thumb range of motion).

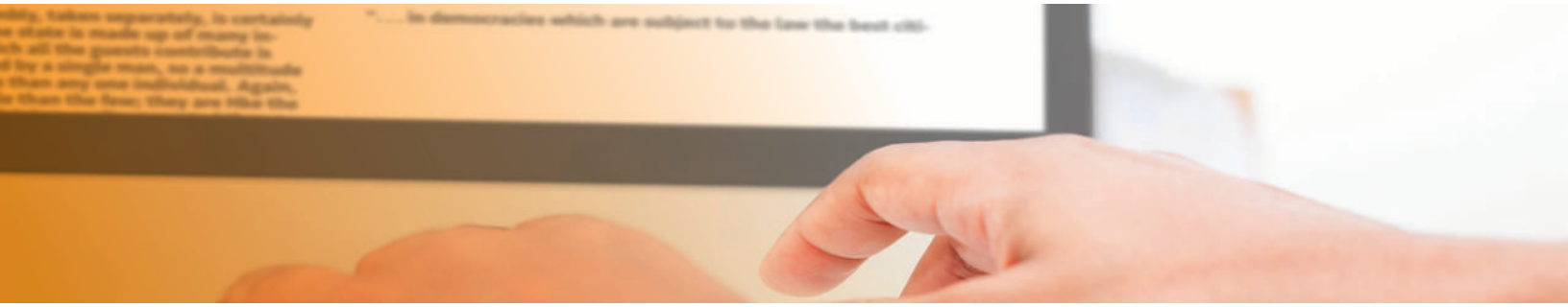


Form Fields

- [ ] On-screen indicators guide the user on what to input into form fields and the coloration and labeling are consistent
- [ ] Form fields are positioned below their labels
- [ ] Virtual keyboard is set to the required type of data entry
- [ ] Methods for data entry are simple and understandable
- [ ] Form fields have screen keyboard, wireless keyboard, and speech data entry capability

Gestures

- [ ] On-screen indicators and accessible text show how to use gestures
- [ ] Touchscreen gestures are simple to perform, and directions are clear
- [ ] Touch controls activate upon release, not initial touch
- [ ] Multiple or alternative forms of gestures are available



Text

- [ ] Text can be resized to at least 200% without using assistive technology
- [ ] Content is presented without loss of information or function when magnified
- [ ] Content does not require zooming in two directions when magnified
- [ ] Browser pinch to zoom is not blocked
- [ ] Font is sans serif to promote readability (for example, Helvetica, Arial, Open Sans)
- [ ] Fonts are limited to two or three variations and are effectively used to build a visual hierarchy

Graphics & Media

- [ ] Graphical content has an accessible text alternative
- [ ] Videos provide text transcripts and audio descriptions
- [ ] Digital text – not images of text – is used when possible
- [ ] Most text has a contrast ratio of at least 4.5:1

Graphics & Media Continued

- [ ] Large text has a contrast ratio of at least 3:1
- [ ] Interface and graphical components have a contrast ratio of at least 3:1
- [ ] Information meaning (for example, an error) is not conveyed by color alone
- [ ] For people who have cataracts (which is normal during aging), design a dark mode to better accommodate them since they perceive dark mode better.
  - Do not use pure black (#000000) and white (#ffffff). Instead, use dark gray as the background and semi-transparent white as the foreground.
  - Be aware of the color contrast ratio of the dark gray and your branding colors. Re-test the color contrast ratio and adjust, as needed. It is possible that the color works well in light mode but does not work in dark mode.

Automated Testing Resources

Common tools are available to check for accessibility, such as the popular Accessibility Scanner. While a few links are provided below, we realize that new resources are being released regularly. We welcome you to try other accessibility tools in the app stores and use whatever works best for your team and your technology in receiving the feedback necessary. As a reminder, these tools are excellent support mechanisms but should not be a replacement for actual user testing in the real world.

- Accessibility Inspector for iOS app
- Android Studio has its internal tool for Android app
- Text and Graphics Color Contrast Checker

References & Resources

- https://www.w3.org/WAI/standards-guidelines/mobile/
- https://www.w3.org/TR/mobile-accessibility-mapping/
- https://uiowa.instructure.com/courses/40/pages/accessibility-principles-pour
- https://www.essentialaccessibility.com/compliance-overview/wcag-web-content-accessibility-guidelines
- https://beta.ada.gov/resources/web-guidance/
- https://blog.usablenet.com/mobile-app-accessibility-techniques-for-inclusive-design-part-1
- https://www.boia.org/definitive-mobile-accessibility-checklist
- https://developer.mozilla.org/en-US/docs/Web/Accessibility/Mobile_accessibility_checklist
- https://pauljadam.com/demos/mobilechecklist.html
- https://www.perkins.org/resource/app-accessibility-checklist/
- https://www.logicsolutions.com/accessibility-checklist-for-your-mobile-app/
- https://www.essentialaccessibility.com/blog/compliance/wcag-for-mobile-apps
- https://a11y-guidelines.orange.com/en/mobile/ios/test/