WAI mobile accessibility: Sprint 1 doc

Links

General quick reference for accessibility: https://www.w3.org/WAI/WCAG21/quickref/
Mobile news page: https://www.w3.org/WAI/standards-guidelines/mobile/
Mobile standards mapping: https://www.w3.org/TR/mobile-accessibility-mapping/

Relevant standards for our app, general web

Text Alternatives

Text alternatives to non-text content:
"All non-text content that is presented to the user has a text alternative that serves the equivalent purpose"
We need to have text or alt text labels for all buttons: including top header to home page, SOS button, clock setting, every option for emergency contacts, etc

Navigation and Pages

Navigation and relationships:
"Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text."
Every header, text section, or clickable icon needs to be navigable and clearly defined in a hierarchy
Ex: each button under each contact should be clearly under that name in a hierarchy, not all flat across the page
Similarly: "If a Web page can be navigated sequentially and the navigation sequences affect meaning or operation, focusable components receive focus in an order that preserves meaning and operability."
Consistency: "Navigational mechanisms that are repeated on multiple Web pages within a set of Web pages occur in the same relative order each time they are repeated, unless a change is initiated by the user."

Multiple ways:
"More than one way is available to locate a Web page within a set of Web pages except where the Web Page is the result of, or a step in, a process."
Let's have a side menu that can jump to any page, not only have to go back to the home page

Keyboard navigation:

"All functionality of the content is operable through a keyboard interface without requiring specific timings for individual keystrokes, except where the underlying function requires input that depends on the path of the user's movement and not just the endpoints”

- [x] Look at what this means for mobile (see below)

"When any component receives focus, it does not initiate a change of context."

No keyboard trap:
"If keyboard focus can be moved to a component of the page using a keyboard interface, then focus can be moved away from that component using only a keyboard interface, and, if it requires more than unmodified arrow or tab keys or other standard exit methods, the user is advised of the method for moving focus away.”

Page Titles:
This will be an important part of the different mobile app pages
"Web pages have titles that describe topic or purpose.”

Location (within pages, not GPS):
"Information about the user's location within a set of Web pages is available.”
Can we make this a toggle display? What is a good way to do this with mobile - side menu with hierarchy?

<u>Input</u>
Multipoint gestures:
"All functionality that uses multipoint or path-based gestures for operation can be operated with a single pointer without a path-based gesture, unless a multipoint or path-based gesture is essential.”
We should not have a need for any multiple-finger gestures - everything should be able to do in one click
Do not expect any need for scrolling or swiping, unless information does not fit in one screen (though mobile design principles probably say it should)

Mobile Actuation:
"Functionality that can be operated by device motion or user motion can also be operated by user interface components and responding to the motion can be disabled to prevent accidental actuation”
We already said we don’t want to do any sort of shaking sensor or function

Concurrent Inputs:
"Web content does not restrict use of input modalities available on a platform except where the restriction is essential, required to ensure the security of the

content, or required to respect user settings.”
When entering text or numbers, users should be able to use the keyboard or input source of their choice, not built into the app

Color and Visual Design
Use of color:
Color is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.
All clickable elements - header sections, menus, and information displayed should have text, alt text, maybe borders
We may want to have a toggle for increased contrast/added borders/different color scheme
A on/off alert should not be only distinguished by red/green - should have text within the image and alt text

Non-text color contrast:
"The visual presentation of the following have a contrast ratio of at least 3:1 against adjacent color(s)”

- **User Interface Components:** Visual information required to identify user interface components and states, except for inactive components or where the appearance of the component is determined by the user agent and not modified by the author;

- **Graphical Objects:** Parts of graphics required to understand the content, except when a particular presentation of graphics is essential to the information being conveyed.

Text spacing
We don’t seem to have a lot of text at the moment, but if we do keep in mind:

- Line height (line spacing) to at least 1.5 times the font size;
- Spacing following paragraphs to at least 2 times the font size;
- Letter spacing (tracking) to at least 0.12 times the font size;

- Word spacing to at least 0.16 times the font size.

Audio or video elements:
Since we have no current video or audio elements, not listing considerations
If we add alert sounds or any gif elements, need to carefully consider these standards
Similarly, we currently don’t have any flashing elements and we need to be careful

if we want any
Consider: will alert flash when it is about to go off or is going off? Let's make sure we don't do this

Focus Visible:
"Any keyboard operable user interface has a mode of operation where the keyboard focus indicator is visible."
We should put a clear border over the elements/buttons that are currently highlighted/active in the navigation

Security and Data
Reauthentication:
"When an authenticated session expires, the user can continue the activity without loss of data after re-authenticating."
This will be relevant if we get into passcodes or other session/security steps

Content
Abbreviations:
"A mechanism for identifying the expanded form or meaning of abbreviations is available"
We should plan to minimize abbreviations regardless

Mobile-specific Standards

WAI does not strongly distinguish between mobile and desktop standards. Part of their explanation:

While mobile is viewed by some as separate from "*desktop/laptop*", and thus perhaps requiring new and different accessibility guidance, in reality there is no absolute divide between the categories. For example:

- many desktop/laptop devices now include touchscreen gesture control,
- many mobile devices can be connected to an external keyboard and mouse,
- web pages utilizing responsive design can transition into various screen sizes even on a desktop/laptop when the browser viewport is resized or zoomed in, and
- mobile operating systems have been used for laptop devices.

Mobile adjustments and mapping:

https://www.w3.org/TR/mobile-accessibility-mapping/
https://w3c.github.io/Mobile-A11y-TF-Note/

Zoom/Magnification:
"A variety of methods allow the user to control content size on mobile devices with small screens."
OS-level features

- Set default text size (typically controlled from the Display Settings) *Note*: System text size is often not supported by mobile browsers.
- Magnify entire screen (typically controlled from the Accessibility Settings). *Note*: Using this setting requires the user to pan vertically and horizontally.
- Magnifying lens view under user's finger (typically controlled from the Accessibility Settings)

Some methods:

- Support for system fonts that follow platform level user preferences for text size.
- Provide on-page controls to change the text size

Contrast:
"Mobile devices are more likely than desktop/laptop devices to be used in varied environments including outdoors, where **glare from the sun** or other strong lighting sources is more likely. This scenario heightens the importance of use of good contrast for all users and may compound the challenges that users with low vision have accessing content with poor contrast on mobile devices."
Suggests: Allow different contrast ratios for larger text

Keyboard control:
Some users will still use a physical keyboard (like a Bluetooth-connected one)
Why keyboards:
Supporting these keyboard interfaces benefits several groups with disabilities:

- People with visual disabilities who can benefit from some characteristics of physical keyboards over touchscreen keyboards (e.g. clearly separated keys, key nibs and more predictable key layouts).
- People with dexterity or mobility disabilities, who can benefit from keyboards optimized to minimize inadvertent presses (e.g. differently shaped, spaced and guarded keys) or from specialized input methods that

emulate keyboard input.
- People who can be confused by the dynamic nature of onscreen keyboards and who can benefit from the consistency of a physical keyboard

Element size and spacing (for clickable elements):
" But these elements must be big enough and have enough distance from each other so that users can safely target them by touch."
Best practices for touch target size include the following:

- Ensuring that touch targets are at least 9 mm high by 9 mm wide.
- Ensuring that touch targets close to the minimum size are surrounded by a small amount of inactive space.

Note: This size is not dependent on the screen size, device or resolution. Screen magnification should not need to be used to obtain this size, because magnifying the screen often introduces the need to pan horizontally as well as vertically, which can decrease usability.

Gesture alternatives:
"Gestures in apps should be as easy as possible to carry out. This is especially important for screen reader interaction modes that replace direct touch manipulation by a two-step process of focusing and activating elements."

Mouseup and touchend events:
Allow users to highlight or read an element without triggering the action, have a specific "fire" event that is not simultaneous with accessing the element

Motion:
Not recommended to do shake or tilt gestures, and we are not planning to do any

Button placement/layout:
Optimize being able to use with one hand
Do we want to move the center buttons to the edges instead?
Do not assume right handedness or thumb use

Screen orientation:
Support both portrait and landscape orientations and do not require the user to rotate the phone at any point (many disabled users have a mounted device or fixed orientation)

Consistent layout:

This is good design anyway, but for the different pages we don't want to get too exciting with variety:
"Components that are repeated across multiple pages should be presented in a consistent layout.'

Important elements before the page scroll:
"If a user with low vision has the screen magnified only a small portion of the page might be viewable at a given time. Placing important elements before the page scroll allows those who use screen magnifiers to locate important information without having to scroll the view to move the magnified area."
We should not have to have a page scroll, with the possible exception of the contacts page or an "additional information about your trip" page
Very important buttons like send or disable alert should be available without scrolling, so maybe at the top of the page - but also consider one-handed touch access from the bottom. We'll have to figure out what is best.

Group elements that perform the same action:
"When multiple elements perform the same action or go to the same destination (e.g. link icon with link text), these should be contained within the same actionable element."
This is important to consider with setting the time. We may want an option to set hours and minutes separately, but we should also allow the user to enter it all in one spot or adjust with one arrow. Similar if they are setting a specific time rather than a duration: they should not *have* to navigate elsewhere to set am/pm - we may want this option but also allow to just click up with the same arrow and automatically turn over. Perhaps also have an option for military time.

Clearly indicate actionable elements:
"Visual users who interact with content using touch or visual cursors (e.g. mice, touchpads, joysticks) should be able to clearly distinguish actionable elements such as links or buttons."
I think this may be an issue with clicking a header for the home screen. This isn't obvious and we should have a "home" button label (in addition to alt text for all headers and visual elements)
Examples of distinguishing features:

1. Conventional shape: Button shape (rounded corners, drop shadows), checkbox, select rectangle with arrow pointing downwards
2. Iconography: conventional visual icons (question mark, home icon, burger icon for menu, floppy disk for save, back arrow, etc)
3. Color offset: shape with different background color to distinguish the element from the page background, different text color

4. Conventional style: Underlined text for links, color for links
5. Conventional positioning: Commonly used position such as a top left position for back button (iOS), position of menu items within left-aligned lists in drop-down menus for navigation

Avoid custom or unusual gestures:
We don't need this and let's not get fancy. The app elements should work as expected in a conventional way.

Data entry/input:
This is important for the variety of information we are asking the user to give us on different points in the process
"Users can enter information on mobile devices in multiple ways such as on-screen keyboard, Bluetooth keyboard, touch, and speech. Text entry can be time-consuming and difficult in certain circumstances. Reduce the amount of text entry needed by providing select menus, radio buttons, check boxes or by automatically entering known information (e.g. date, time, location)."

Small screen layout:
Some best practices for helping users to make the most of small screens include

- Minimizing the amount of information that is put on each page compared to desktop/laptop versions by providing a dedicated mobile version or a responsive design:
    - a dedicated mobile version contains content tailored for mobile use. For example, the content may contain fewer content modules, fewer images, or focus on important mobile usage scenarios.
    - a responsive design contains content that stays the same, but CSS stylesheets are used to render it differently depending on the viewport width. For example, on narrow screens the navigation menus may be hidden until the user taps a menu button.
- Providing a reasonable default size for content and touch controls (see "B.2 Touch Target Size and Spacing") to minimize the need to zoom in and out for users with low vision.
- Adapting the length of link text to the viewport width.
- Positioning form fields below, rather than beside, their labels (in portrait layout)

Another thing to consider: How easy is it to set a complete alert (the whole process) on just one page? Let's look at our UX design to prioritize this. Perhaps

we want to have an option to add additional info (like trail or person info) but the basic functions should be able to do fast and without navigating multiple pages or scrolling significantly (*some* scrolling may be ok but it should also have the option to page down with an arrow/keyboard)