

# Summary and Checklist: iOS Application Accessibility

## Summary

The iOS platform is rich with accessibility possibilities. It does have some limitations in comparison to the accessibility capabilities of full-featured desktop systems, but it comes with the advantage of being able to carry around an accessible mobile computer in your pocket or hand bag. Mobile device accessibility is one of the most powerful enablers of independence for people with disabilities, but only if the apps have been designed with accessibility in mind. It is up to the design and development team to take advantage of the accessibility possibilities in the iOS development environment.

## Checklist

### User Interaction

#### Touch Target Size

- The click/touch target size **SHOULD** be large enough (without using zoom) to facilitate easy use with a finger, without risking activating an adjacent link or button.

#### Target Spacing and Grouping

- Touch targets **MUST NOT** overlap.
- Compact touch targets **SHOULD NOT** be immediately adjacent to each other.
- Active elements with the same target **SHOULD** be grouped into one touch target.

#### Activate on Release

- Touch controls **MUST NOT** activate when an item is first touched.

#### Kinetic/Accelerometer

- An application **MUST NOT** rely on kinetic motion (detected by the accelerometer) alone.
- The user **SHOULD** be able to disable motion-sensitive features.

#### Voice

- An application **MUST NOT** rely on voice control alone.

#### Gestures

- An application **MUST NOT** rely on gestures alone.
- Whenever possible, actions **SHOULD** be mapped to their corresponding default iOS gestures.

## UI Controls

### Buttons

- A button **MUST** always convey its role using the correct trait.
- An active image used as a button **MUST** convey the correct role.
- When a button is disabled it **MUST** convey its state.

### Links

- The role and purpose of a link **MUST** be clearly conveyed by VoiceOver.

### Progress Bar

- Progress bars **MUST** convey their visible name to VoiceOver users with a descriptive, unique accessibility label.
- Progress spinners (UIActivityIndicatorView) **MUST** speak correct alternative text or immediately receive focus in order to convey an accessible name to VoiceOver users.

### Slider

- Slider controls **MUST** have a visible label that is programmatically associated using the accessibility label property.

### Switch

- Switch controls **MUST** have a visible label that is programmatically associated using the accessibility label property.

## Notifications

### Notifying Users of Changes

- Dynamically generated content following user interaction **MUST** meet one of the following: assistive technology is automatically made aware of the new content **OR** the new content is the very next thing the assistive technology will encounter on the page.

### Time Limits

- If there is a session time limit, users **MUST** be warned before the session ends and **MUST** be given time to save their data and/or extend the session.
- Incomplete data **SHOULD** be saved after a session timeout.

### Alert

- Alerts **MUST** include a title and description.

# Summary and Checklist: iOS Application Accessibility

## Notifications (continued)

### Dialog

- Custom Dialogs MUST set `accessibilityViewIsModal` to "true" and the focus MUST be sent to the dialog.

## Custom Elements

### Accordion

- Accordion components MUST announce their accessible name and their expanded or collapsed state.

### Autocomplete

- VoiceOver users MUST be automatically made aware of the autocomplete suggestions when they appear.

### Rating

- A rating control MUST always convey its current value, any visible information, and be operable by both VoiceOver and non-VoiceOver users.

### Sortable Table

- If a table is sortable, the header cells' accessibility labels MUST dynamically update to correctly convey their sorted state each time the data order changes.

### Toggle

- Toggle buttons MUST convey an accessible name with programmatically associated, visible labels and convey their state using the accessibility label.
- Image toggle buttons MUST have descriptive, accurate alternative text provided in the accessibility label.

## Form Labels and Validation

### Labels

- The label MUST be visible at all times.
- Labels MUST be programmatically associated with their corresponding control elements.
- A "hint" SHOULD be provided when the label doesn't sufficiently convey a control's purpose to VoiceOver users.

### Group Labels

- For groups of related controls where the individual labels for each control do not provide a sufficient description, additional group labels MUST be provided.

### Checkboxes

- Checkboxes MUST convey their selected or checked state to VoiceOver.

### Checkboxes (continued)

- Groups of checkboxes MUST provide a common group label as well as a unique label for each control via their accessibility label properties.

### Date Picker

- A standard Date Picker control SHOULD be used instead of a text field in order to avoid forcing users to enter a date with a default keyboard.

### Radio Buttons

- Radio buttons MUST provide a common group label as well as a unique label for each control via their accessibility labels and they MUST convey their selected state.

### Select (UIPickerView)

- Controls using UIPickerView MUST be programmatically associated with their visible label in order to provide an accessible name.

### Text Fields

- Text fields MUST have programmatically associated, visible labels.

### Validation and Error Reporting

- Efficient, intuitive, and clearly identified text based alerts MUST be provided to users for form validation cues and errors.
- If input errors are automatically detected, suggestions MUST be provided in text for fixing the input in a timely and accessible manner before the data is submitted to the server.

## Navigation

### Page Titles

- Screens MUST have meaningful titles.

### Tab Panel

- Tabs MUST convey their role, their current state, and their order in the total number of tabs to VoiceOver users.

### Menu

- Menu items MUST have the appropriate trait in order to convey their accessible role.
- The button that opens the menu MUST have a clear, descriptive title or accessibility label to accurately convey its purpose.
- VoiceOver users MUST be able to dismiss the menu.
- When the menu is closed, focus MUST return to the triggering button.

# Summary and Checklist: iOS Application Accessibility

## Device Orientation

- The orientation of the content **MUST** not be locked to either landscape or portrait unless a specific orientation is essential for the functionality of the app.

## Semantic Structure and Meaning

### Headings

- Headings used to title or describe sections of content **MUST** use the header trait in order to convey their semantic purpose.

### Tables

- Data tables **MUST** have the correct reading order, **MUST** convey the row and column header text for each data cell, and **MUST** correctly associate the data cell with the header cell in the accessibility label.

### Reading Order and Focus Order

- The reading order of a screen as conveyed by VoiceOver **MUST** be logical and intuitive.
- When content is added or deleted from a screen, VoiceOver focus **MUST** be managed logically and never be lost.

## Color and Contrast

### Colors that Convey Information

- Any information conveyed by color **MUST** be accompanied by a programmatically-discernible text alternative.
- The text alternative for information conveyed by color **MUST** accurately convey the same information without color.
- Any information conveyed by color **MUST** be accompanied by a visible alternative (text, image, etc.) that does not depend on color for meaning.
- Color alone **MUST NOT** be used to distinguish links from surrounding text unless the color contrast between the link and the surrounding text is at least 3:1. An additional differentiation (e.g. underline, outline, etc.) **SHOULD BE** provided as well.

## Contrast

- Small text (under 18 point regular font or 14 point bold font) **MUST** have a contrast ratio of at least 4.5 to 1 with the background.
- Large text and images of large text (at or over 18 point or 14 point bold) **MUST** have a contrast ratio of at least 3 to 1 with the background.
- The contrast of UI control boundaries compared to adjacent areas **SHOULD** be sufficient (3 to 1 for UI control boundaries measuring at least 3px by 3px, and 4.5 to 1 for all other UI controls) to distinguish the UI control from the adjacent areas.
- Small text (under 18 point regular font or 14 point bold font) **SHOULD** have a contrast ratio of at least 7 to 1 with the background.

## Animations and Motion

### Parallax and Accelerometer Animations

- Moving the physical device **SHOULD NOT** cause non-essential visual motion or animations (e.g. parallax effects)
- Users **MUST** be able to disable any non-essential motion-sensitive visual effects.

### Moving or Disappearing Content

- Automatically moving, blinking, or scrolling content that lasts longer than 5 seconds **MUST** be able to be paused, stopped, or hidden by the user.

### Blinking or Flashing Content

- A screen **MUST NOT** contain content that flashes more than 3 times per second unless that flashing content is sufficiently small, the flashes are of low contrast, and do not violate general flash thresholds.

## Audio and Video

### Captions

- All prerecorded video **MUST** have synchronized captions.
- All live multimedia (video plus audio) events that contain dialog and/or narration **MUST** be accompanied by synchronized captions.
- Live audio consisting mainly of dialog and/or narration **SHOULD** have synchronized captions.

### Transcripts

- All prerecorded video and audio **MUST** have text transcripts.

# Summary and Checklist: iOS Application Accessibility

## Text Content

### Language Settings

- The language of the page **MUST** be programmatically identified.
- The language of sections of content that are different from the app's default language **MUST** be identified in such a way that they can be discovered by assistive technology.

### Text Resize

- The screen **MUST** be readable and functional when text is set to 200% of its initial size.

### Emojis

- Emojis **MAY** be used in text content.
- Custom emojis in custom keyboards **MUST** have alternative text.

## Images

### Alternative Text

- Images that convey information **MUST** have a descriptive accessibility label that serves the same purpose and presents the same information as the image.
- Images that are active **MUST** have an accessibility label that describes the purpose or function of the image.
- Images that are too complex to be fully described in a short text alternative **MUST** have their purpose described using an extended text alternative.
- Images that do not convey content, are decorative, or with content that is already conveyed in the text **MUST NOT** be given an alternative text equivalent that is exposed to assistive technology.

### Text in Images

- An image **MUST NOT** include informative text if an equivalent visual presentation of the text can be rendered using real text.