POWERAPPS ACCESSIBILITY STANDARDS AND GUIDELINES

AS OF MARCH 09, 2019

Summary of the Microsoft White paper

Link to source document

“An accessible PowerApps canvas app begins by having an orderly, consistent way of defining the controls on each screen.

Some controls are decorative, like a horizontal line between Form sections, or important but non-interactive, like text on a label.”

1 - KEYBOARD ACCESSIBILITY

- ✓ Have a control flow that matches your language. Ex: Move the focus left to right for English speakers
- ✓ Set the TabIndex property to 0 or greater
- ✓ Make the focus border visible:
  - Set the FocusBorderColor property to a minimum 3:1 contrast ratio with the background color
  - Set the FocusBorderThickness property to 0 or greater
  - Set the BorderStyle property to BorderStyle.Dotted if the control doesn't have a border when it is unfocused

2 - SCREEN

- ✓ Make the screen names understandable when announced by a screen reader.
- ✓ End all the screen names with the word “screen” so it’s clear what is being announced
- ✓ The screen name should describe the screen’s contents and function (Ex: Search Colleagues Screen)
- ✓ Create a separate screen for pop-ups or dialog box instead of having it overlay on an existing screen

3 -LABELS

- ✓ Set the Visible property to True for Labels, so that screen readers announce the label control's Text property.
- ✓ Make sure text has a minimum luminosity contrast ratio of 4.5:1 against the background
- ✓ Set the role property for labels correctly
  - Screen headings --> Heading1
  - Subheadings and gallery names ---> Heading2 and Heading3
  - Gallery item names --> Heading4 roles
- ✓ Use **live labels** to enable screen readers to announce events such as changes to the label's Text property, the selection of an item or when an item is inserted or deleted from the UI.
- ✓ Use Button controls instead of interactive labels (Labels that have the On Select property set to perform an action)

3 -LABELS

LIVE LABELS

Labels have a Live property that can enable screen readers to announce changes to the label's Text property. This property is useful for announcing dynamic changes in the app's UI in an accessible way. The live property has three settings, and each behaves differently:

OFF SCREEN READERS

Off Screen readers don't announce dynamic changes

POLITE SCREEN READERS

Polite Screen readers finish speaking before announcing any changes that occurred while speaking. the screen reader announces the Text property of the label as it changes

ASSERTIVE SCREEN READERS

Assertive Screen readers interrupt themselves to announce any changes that occur while speaking. the screen reader announces dynamic changes in the app— that is, it announces the Text property of the label as it changes

4 - BUTTONS

- ✓ Set the Text property be indicative of the button's action (because screen readers announce the Text property of a button)
- ✓ Use tooltips to provide additional information to the button's Text property, such as its selected state when the user moves the mouse pointer over the button
- ✓ Set the DisplayMode property to Disabled for all disabled buttons so that the screen reader explicitly announces that the button is disabled
- ✓ Set visible text to have a minimum luminosity contrast ratio of 4.5:1 against the background
- ✓ Make the button keyboard accessible

5 - TEXT INPUT

- ✓ Use the AccessibleLabel property instead of the HintText property because different browsers and screen readers handle this differently
- ✓ Set visible text must have a minimum luminosity contrast ratio of 4.5:1 against the background
- ✓ Make the AccessibleLabel property descriptive so that the user can understand the purpose of the Text Input control
- ✓ Announce the maximum number of characters to your users by setting the AccessibleLabel property appropriately. Notify users as the content of the Text Input control approaches the maximum character limit

6 - HTML TEXT

- ✓ Set visible text must have a minimum luminosity contrast ratio of 4.5:1 against the background
- ✓ Ensure that HTML Text controls do not contain interactive elements like <button>, <a>, or <input> because The TabIndex system in PowerApps does not consider elements inside HTML text.

7 - GALLERY

- ✓ Be sure to make the AccessibleLabel property descriptive of the gallery's contents because the text in the AccessibleLabel property of the Gallery control is announced by screen readers.

- ✓ Enable the ItemAccessibleLabel property in order for screen readers to announce the state of each menu as "Selected" or "not selected". This can be done with the ItemAccessibleLabel property:

    - Set the Gallery control's Selectable property to true.
    - Set the ItemAccessibleLabel property to something similar to: If(ThisItem.IsSelected=true, “Selected”, "Not selected")

8 - IMAGE

- ✓ Set interactive images to have a minimum of a 3:1 contrast ratio against adjacent colors
- ✓ Images that are purely decorative have no minimum contrast requirement.
- ✓ Describe the purpose of the image in the AccessibleLabel property (for interactive Image controls only)

9 - ICONS

- ✓ Set interactive Icons to have a minimum of a 3:1 contrast ratio with the adjacent colors
- ✓ Icons should have tooltips for understanding and operating content
- ✓ Describe the purpose of the image in the AccessibleLabel property (for interactive Icon controls only)
- ✓ Use icons instead of images wherever possible because the PowerApps platform ensures that icons are visible in high-contrast mode

10 - ADD PICTURE

- ✓ Ensure that screen readers announce when a picture is added, changed, or deleted. This can be done using a live label

- ✓ Use descriptive text in the Text property to help users understand the context.

11 - DATACARD

- ✓ Ensure that screen readers announce when a picture is added, changed, or deleted. This can be done using a live label
- ✓ Use descriptive text in the Text property to help users understand the context.

12 - CHECKBOX

- ✓ The PowerApps platform ensures that screen readers will announce the current state of the check box—whether it is checked or unchecked.
- ✓ There must be a minimum of 3:1 color contrast ratio between the values for:
  - CheckmarkFill and CheckboxBackgroundFill
  - CheckboxBackgroundFill and Fill
  - CheckboxBackgroundFill and PressedFill
  - CheckboxBackgroundFill and HoverFill
- ✓ The PowerApps platform ensures that check boxes are visible in high-contrast mode.
- ✓ As changes are made to the Check Box control value, the PowerApps platform ensures that the new state is announced by the screen reader.

13 - SLIDERS

- ✓ The PowerApps platform will ensure that the screen reader announces the value of the slider and any changes to it
- ✓ The PowerApps platform ensures that sliders are visible in high-contrast mode

14 - TOGGLES

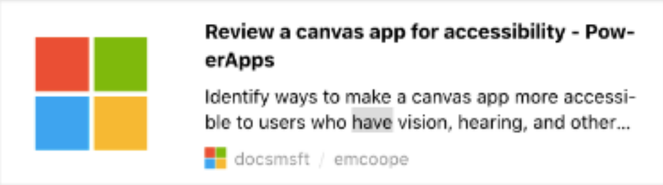
- ✓ Screen readers will announce the toggle value
- ✓ Screen readers must also announce any change made to the toggle value. This can be done by using a live label
- ✓ The PowerApps platform ensures that toggles are visible in high-contrast mode

15 - TIMER

- ✓ Timer controls are typically hidden and used to manage some kind of event
- ✓ If the timer is a visible interactive control on the screen, you can make it keyboard accessible

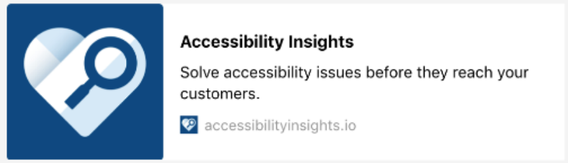
ACCESSIBILITY TESTING

1 - THE ACCESSIBILITY CHECKER



After you have built your app, be sure to check it using the Accessibility checker, which scans your app for common issues such as uncaptioned images and controls that are missing the TabIndex property

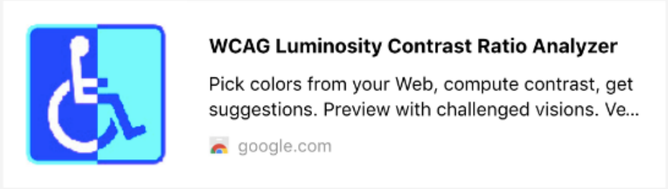
2 - ACCESSIBILITY INSIGHTS



Accessibility Insights is an accessibility testing tool for makers. It helps makers to find accessibility issues during development, before they reach your users

This helps the maker know how each control will be called out. It results in fewer bugs in the testing phase. Makers can even record the tests done in accessibility insights

3 - COLOR CONTRAST ANALYZER



Feature in Accessibility Insights that helps investigate contrast ratios.

Large-scale text and images of large-scale text that have a contrast ratio of at least 4.5:1 pass the color contrast test

4 - ASSISTIVE TECHNOLOGIES

ANDROID TALKBACK

NARRATOR (SCAN MODE)

VOICEOVER

We recommend using Microsoft Edge while working with Narrator. Chrome and Firefox do not work well with Narrator, and Internet Explorer uses an older accessibility API

Note: The Tooltip property for a label is read inconsistently by different screen readers and browsers. For example, Narrator will read a tooltip after a long pause, but other screen readers might not announce it at all. Some screen readers allow the user to toggle reading on and off. For these reasons, don’t rely on the Tooltip property for accessibility

5 - HIGH CONTRAST MODE

- ✓ Makers need to ensure that all controls are visible in high-contrast mode
- ✓ Recommendations:
  - If there are overlapping controls on a screen, ensure that the controls are reordered by using Bring Forward or Send Backward in such a way that all the controls are visible in high-contrast mode. This can be done by right-clicking the control and selecting the appropriate option under the Reorder tab
- ✓ For informative or functional images, we recommend using vector formats like SVG images instead of raster formats like PNG images for the accessible high-contrast mode. Decorative images have no accessibility requirements