WCAG 2.2
Made Simple

A Plain-Language Guide to Web Accessibility

For FDOT Staff
Content Creators • Designers • Developers

Office of Information Technology
January 2026

Table of Contents

What Is This Guide?

This guide translates the Web Content Accessibility Guidelines (WCAG) 2.2 into plain language that anyone can understand and apply. Whether you create documents, design web pages, or develop applications for FDOT, this resource will help you make your work accessible to everyone.

> ⚠ **April 24, 2026 Deadline**
>
> The U.S. Department of Justice requires all state and local government digital content to meet WCAG 2.1 Level AA standards by this date. WCAG 2.2 (released October 2023) builds on 2.1 with important improvements. This guide covers both to help FDOT exceed minimum requirements.

The Four Principles at a Glance

WCAG is organized around four principles. All accessibility requirements fall under one of these:

1. PERCEIVABLE	2. OPERABLE	3. UNDERSTANDABLE	4. ROBUST
Can users see, hear, or access the content?	Can users navigate and interact with the interface?	Can users understand the content and how to use it?	Does the content work with assistive technologies?
• Alt text • Captions • Contrast • Structure	• Keyboard • No traps • Timing • Navigation	• Language • Predictable • Error help • Labels	• Valid code • Name/role • Status msgs • Future-proof

Understanding WCAG Levels

WCAG organizes requirements into three conformance levels:

Level	What It Means	For FDOT
Level A	Minimum requirements. Content failing Level A is seriously inaccessible.	Must meet all Level A criteria
Level AA	The standard for compliance. What most laws require, including the DOJ rule.	**FDOT's TARGET — Required by April 2026**
Level AAA	Enhanced accessibility. Not all content can meet AAA, but aim for it when possible.	Best practice — implement where feasible

Who Benefits from Accessible Content?

Accessibility helps everyone, but it is essential for people who:

- Use screen readers (people who are blind or have low vision)
- Navigate with a keyboard only (people with motor disabilities)
- Have difficulty seeing colors or low contrast (color blindness, low vision)
- Need captions for audio (people who are deaf or hard of hearing)
- Have cognitive or learning differences
- Use mobile devices, have slow connections, or work in challenging environments

Principle 1: Perceivable

Users must be able to perceive the information being presented. Content cannot be invisible to all of their senses.
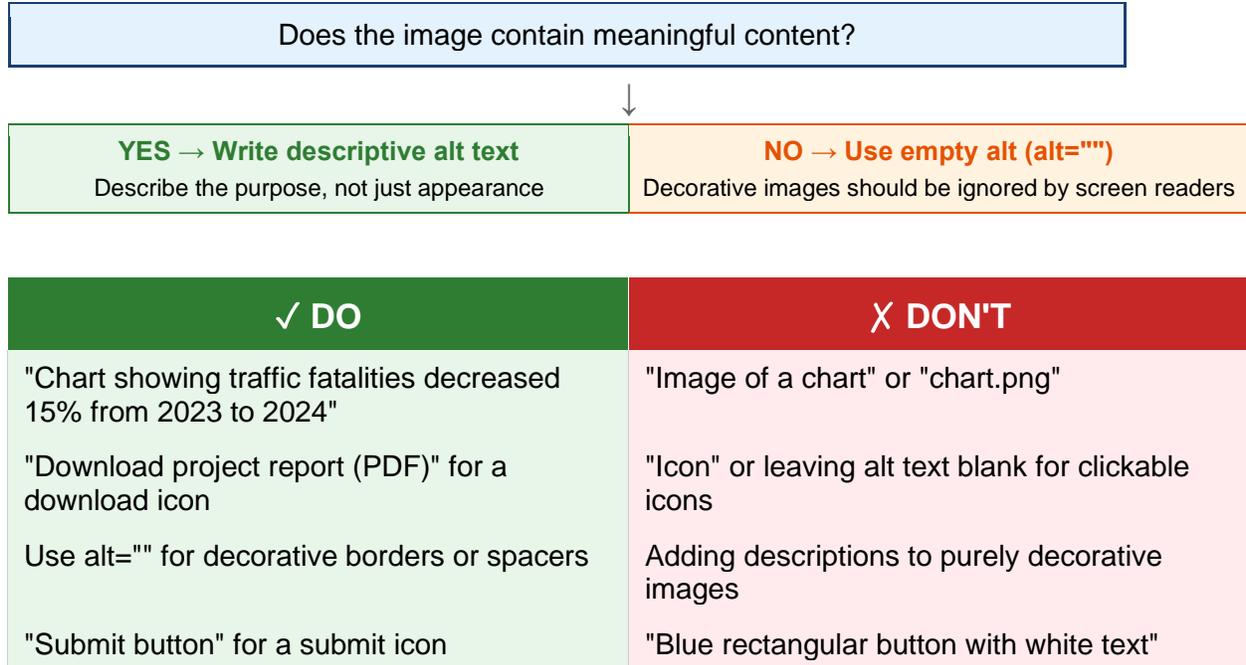
> 💡 **The Key Question**
>
> If someone can't see an image, hear audio, or distinguish colors, can they still get the same information another way?

1.1 Text Alternatives (Images)

Every image needs a text description so screen readers can convey the information to users who cannot see it.

Alt Text Decision Guide

Use this decision process when adding alt text to any image:

Does the image contain meaningful content?

↓

YES → Write descriptive alt text	NO → Use empty alt (alt="")
Describe the purpose, not just appearance	Decorative images should be ignored by screen readers

✓ DO	✗ DON'T
"Chart showing traffic fatalities decreased 15% from 2023 to 2024"	"Image of a chart" or "chart.png"
"Download project report (PDF)" for a download icon	"Icon" or leaving alt text blank for clickable icons
Use alt="" for decorative borders or spacers	Adding descriptions to purely decorative images
"Submit button" for a submit icon	"Blue rectangular button with white text"

1.2 Time-Based Media (Audio & Video)

Videos and audio content need alternatives so everyone can access the information.

Captions (Required for all videos): Accurate, synchronized text of all spoken content and important sounds. Auto-generated captions must be reviewed and corrected.

Audio Descriptions (Level AA): Narration describing important visual information not conveyed in the audio track.

Transcripts: Text versions of audio/video content for users who prefer reading or cannot access media.

1.3 Adaptable Content (Structure)

Content must be structured so it can be presented in different ways without losing information.

Heading Structure

Use heading levels (H1, H2, H3) to create a logical outline. Think of headings like a table of contents—they help screen reader users navigate.

✓ DO	✗ DON'T
Use one H1 per page for the main title	Use multiple H1 headings on one page
Use H2 for major sections, H3 for subsections	Skip heading levels (H1 → H3)
Keep heading levels in order (H1 → H2 → H3)	Use headings just to make text big or bold

1.4 Distinguishable (Color & Contrast)

Never Rely on Color Alone

About 8% of men have some form of color blindness. Never use color as the only way to convey information.

✓ DO	✗ DON'T
"Required fields are marked with * and shown in red"	"Click the green button" without another identifier
Use patterns AND colors in charts and graphs	Use only red/green to show pass/fail status
Underline links (don't rely only on blue color)	Make links distinguishable only by color

Contrast Requirements

- Normal text: 4.5:1 minimum contrast ratio
- Large text (18pt+ or 14pt+ bold): 3:1 minimum
- UI components and graphics: 3:1 minimum

> 💡 **Free Tool**
>
> Use WebAIM's Contrast Checker at webaim.org/resources/contrastchecker to test your color combinations before publishing.

Principle 2: Operable

Users must be able to operate the interface. The interface cannot require interactions that a user cannot perform.

> ## ⚲ The Key Question
>
> Can someone use your content with just a keyboard? Can they take their time? Will anything cause a seizure?

2.1 Keyboard Accessible

Everything that can be done with a mouse must also work with a keyboard.

Tab Order: Users should be able to press Tab to move through all interactive elements in a logical order.

No Keyboard Traps: Users must be able to move away from any element using standard keys. Never trap focus.

Focus Indicators: There must always be a visible indicator showing which element has keyboard focus.

> ## ⚲ Quick Test
>
> Put your mouse aside and try to complete all tasks on your page using only Tab, Enter, and Arrow keys. Can you reach everything?

2.2 Enough Time

- If a time limit is necessary, allow users to extend, adjust, or turn it off
- Provide controls to pause, stop, or hide moving content
- Avoid auto-playing carousels; let users control the pace

2.3 Seizures and Physical Reactions

> ## ⚠ Critical Safety Rule

Nothing should flash more than 3 times per second. Flashing content can trigger seizures in people with photosensitive epilepsy.

2.4 Navigable

Page Titles: Every page needs a unique, descriptive title in the browser tab.

Skip Links: Provide "Skip to main content" so keyboard users can bypass navigation.

Descriptive Link Text: Links should make sense out of context. Avoid "click here" or "read more."

Focus Not Obscured (New in WCAG 2.2)

When an element receives keyboard focus, it must not be completely hidden by sticky headers, footers, or other fixed content.

2.5 Input Modalities

Target Size (New in WCAG 2.2)

Interactive elements must be large enough to tap or click easily:

- Minimum: 24×24 CSS pixels (Level AA)
- Recommended: 44×44 CSS pixels (Level AAA)

Dragging Movements (New in WCAG 2.2)

If your interface uses drag-and-drop, you must provide an alternative. For example, if users can drag items to reorder a list, also provide up/down arrow buttons.

Principle 3: Understandable

Users must be able to understand the information and the operation of the interface.

> ### 💡 The Key Question
>
> Can people understand your content and figure out how to use your forms and navigation without confusion?

3.1 Readable

Language of Page: Set the language in HTML (e.g., lang="en") so screen readers pronounce words correctly.

Language of Parts: Mark content in different languages (e.g., a Spanish phrase in an English document).

Plain Language: Write clearly. Define jargon and abbreviations. Aim for 8th-grade reading level when possible.

3.2 Predictable

No Surprises: When elements receive focus or input changes, don't automatically navigate away or submit forms.

Consistent Navigation: Navigation menus should appear in the same location and order on every page.

Consistent Help (New in 2.2): If you provide help mechanisms, put them in the same place on every page.

3.3 Input Assistance (Forms)

- Label every form field clearly (don't rely only on placeholder text)
- Mark required fields with text, not just an asterisk or color
- Provide clear error messages that explain what's wrong and how to fix it
- Group related fields together with clear labels

Redundant Entry (New in WCAG 2.2)

Don't make users re-enter information they've already provided. Auto-fill previous entries or let users select from a list.

Accessible Authentication (New in WCAG 2.2)

Login processes shouldn't require cognitive tests like memorizing passwords or solving CAPTCHAs without alternatives.

✓ DO	✗ DON'T
Allow password managers to fill credentials	Use CAPTCHAs without alternatives
Offer copy-paste for verification codes	Block password managers from working
Provide email link or biometric alternatives	Require users to retype passwords to confirm

Principle 4: Robust

Content must be robust enough to work with current and future technologies, including assistive technologies.

> **💡 The Key Question**
>
> Does your code work correctly with screen readers and other assistive technologies?

4.1 Compatible

Name, Role, Value: Every interactive component must have a name assistive technology can read, a role describing what it is, and a value/state when applicable.

- Use native HTML elements when possible (button, a, input)
- If creating custom widgets, use ARIA roles and properties correctly
- Test with actual screen readers, not just automated tools

Status Messages: When content changes (like a success message), notify users without moving their focus. Use ARIA live regions.

What's New in WCAG 2.2

WCAG 2.2 was released in October 2023 and adds 9 new success criteria. Here's a summary of the most impactful additions for FDOT:

New Criterion	Level	What It Means
Focus Not Obscured (Minimum)	**AA**	Focused elements can't be hidden by sticky headers/footers
Dragging Movements	**AA**	Provide alternatives to drag-and-drop interactions
Target Size (Minimum)	**AA**	Interactive targets must be at least 24×24 pixels
Consistent Help	**A**	Help mechanisms appear in the same location on each page
Redundant Entry	**A**	Don't require re-entry of previously provided info
Accessible Authentication (Minimum)	**AA**	No cognitive tests for login without alternatives

> 💡 **Note About Parsing (4.1.1)**
>
> WCAG 2.2 removed the "Parsing" criterion because modern browsers handle HTML errors gracefully. You no longer need to worry about this specific criterion.

Quick Reference Checklists

For Content Creators & Editors

- ☐ Add meaningful alt text to all images
- ☐ Use proper heading structure (H1, H2, H3)
- ☐ Write descriptive link text (not "click here")
- ☐ Add captions to all videos
- ☐ Use plain language and define jargon
- ☐ Don't use color alone to convey meaning
- ☐ Make PDFs accessible (tags, reading order, alt text)

For Designers

- ☐ Ensure 4.5:1 contrast ratio for text (3:1 for large text)
- ☐ Make touch targets at least 24×24 pixels
- ☐ Design visible focus indicators
- ☐ Don't rely on color alone for information
- ☐ Ensure content works at 200% zoom
- ☐ Ensure sticky elements don't cover focused content

For Developers

- ☐ Use semantic HTML (proper headings, lists, landmarks)
- ☐ Make all functionality keyboard accessible
- ☐ Never remove focus outlines without replacement
- ☐ Set the page language in HTML (lang attribute)
- ☐ Use ARIA only when native HTML isn't sufficient
- ☐ Provide skip navigation links
- ☐ Associate form labels with inputs programmatically
- ☐ Provide alternatives to drag-and-drop
- ☐ Allow password managers in login forms

Testing Your Work

Quick Manual Tests

1. **Keyboard Test:** Navigate your entire page using only Tab, Enter, and Arrow keys
2. **Zoom Test:** Zoom to 200% and check that all content is still usable
3. **Heading Test:** Use a browser extension to view heading structure
4. **Color Test:** View your page in grayscale—is information still clear?
5. **Screen Reader Test:** Use NVDA (free) or VoiceOver (Mac/iOS) to hear your content

> ⚠ **Important**
>
> Automated tools catch only about 30-40% of accessibility issues. Manual testing with keyboards and screen readers is essential.

Free Testing Tools

- **WAVE** (wave.webaim.org) — Browser extension for quick checks
- **axe DevTools** — Browser extension for detailed technical testing
- **WebAIM Contrast Checker** — Test color contrast ratios
- **NVDA** — Free screen reader for Windows
- **Siteimprove** — Available to FDOT for comprehensive scanning

Additional Resources

- WCAG 2.2 Quick Reference: w3.org/WAI/WCAG22/quickref/
- Yale Accessibility Articles: usability.yale.edu/web-accessibility/articles
- WebAIM: webaim.org
- DOJ ADA Web Rule: ada.gov/resources/web-rule-first-steps/