Accessible forms checklist

28 WCAG 2.2 success criteria at levels A and AA that matter on a form. Tick each one that passes. Level AA includes every Level A item.

Form: _____ Tested by: _____ Date: _____

Quick test, about 20 minutes

Keyboard: put the mouse away. Fill in and submit the form with Tab, Shift+Tab, Enter, Space and arrows.

Errors: submit it empty, then with a bad email. Read every message out loud.

Zoom: 200 percent, then 400 percent on a laptop window. Nothing should be cut off or scroll sideways.

Phone: fill it in with one thumb. Every target should be easy to hit.

Screen reader: NVDA (free, Windows) or VoiceOver (built into Apple devices). Listen to each label.

Checker: run an automated checker too. It finds some problems, not all.

Labels and instructions

- ☐ 3.3.2 Labels or Instructions (A). Every field has a visible label. Formats (like a date) and required fields are explained before people type.
- ☐ 1.3.1 Info and Relationships (A). Each label is tied to its field in the code. Related choices sit in a group with a group name.
- ☐ 2.4.6 Headings and Labels (AA). Labels and page headings say what each field or section is for.
- ☐ 2.5.3 Label in Name (A). The name a screen reader announces contains the visible label text.
- ☐ 1.1.1 Non-text Content (A). Images and picture choices have a text alternative. Any CAPTCHA offers another way through.
- ☐ 1.3.5 Identify Input Purpose (AA). Fields about the person (name, email, phone, address) tell the browser what they are, so autofill works.

Required fields and errors

- ☐ 3.3.1 Error Identification (A). When an answer is missing or wrong, the field is identified and the error is described in text.
- ☐ 3.3.3 Error Suggestion (AA). Error messages say how to fix the problem, for example the format you expect.
- ☐ 1.4.1 Use of Color (A). Errors and required fields are not shown by colour alone. Add words or a symbol with a text key.
- ☐ 4.1.3 Status Messages (AA). Messages like "Two answers need fixing" are announced by screen readers without moving focus.
- ☐ 3.3.4 Error Prevention (Legal, Financial, Data) (AA). Forms that take money, make a legal commitment or change stored data let people check and correct answers, or undo, before the final step.

Keyboard and focus

- ☐ 2.1.1 Keyboard (A). Every field, choice, date picker, upload and button works with a keyboard alone.
- ☐ 2.1.2 No Keyboard Trap (A). Focus can always move out of any field or widget with the keyboard.
- ☐ 2.4.3 Focus Order (A). Focus moves in an order that keeps the meaning, usually top to bottom.
- ☐ 2.4.7 Focus Visible (AA). You can always see which field or button has keyboard focus.

- ☐ 2.4.11 Focus Not Obscured (Minimum) (AA, new in 2.2). A focused field is never fully hidden by a sticky header, a cookie banner or a chat button.
- ☐ 3.2.2 On Input (A). Picking an option or typing does not submit the form or open a new page without warning.
- ☐ 2.2.1 Timing Adjustable (A). If the form has a time limit, people can turn it off, adjust it or extend it.

Colour and contrast

- ☐ 1.4.3 Contrast (Minimum) (AA). Text, labels and button text reach a contrast ratio of 4.5:1 (3:1 for large text).
- ☐ 1.4.11 Non-text Contrast (AA). Field borders, checkboxes, radio buttons and focus rings reach 3:1 against the colours next to them.

Size, zoom and touch

- ☐ 1.4.4 Resize Text (AA). Text can grow to 200 percent without cutting off content or controls.
- ☐ 1.4.10 Reflow (AA). At 320 CSS pixels wide, the form fits with no sideways scrolling.
- ☐ 2.5.8 Target Size (Minimum) (AA, new in 2.2). Buttons, checkboxes and other targets are at least 24 by 24 CSS pixels, or have enough space around them.

Language

- ☐ 3.1.1 Language of Page (A). The page declares the form's language.
- ☐ 3.1.2 Language of Parts (AA). Words or passages in another language are marked with their own language.

Fewer repeats and puzzles

- ☐ 3.3.7 Redundant Entry (A, new in 2.2). Answers given earlier in the same process are filled in or offered, not typed again.
- ☐ 3.3.8 Accessible Authentication (Minimum) (AA, new in 2.2). A login step needs no memory test or puzzle unless another way exists. Password managers and paste work.

Custom controls

- ☐ 4.1.2 Name, Role, Value (A). Custom widgets tell assistive technology their name, role and state (checked, selected, expanded).

Criterion numbers, names and levels: W3C, Web Content Accessibility Guidelines (WCAG) 2.2, https://www.w3.org/TR/WCAG22/ (accessed 28 October 2026). The short descriptions are plain-language summaries; the W3C text is the full rule. The xlsx version of this checklist adds how to test each item and counts passes and fails.