Buttons

What is it for?

Buttons let people take an action, such as submitting a form, saving information or opening a menu. They are one of the most common interactive elements, so if a button isn't accessible, people can be completely blocked from finishing a task.

Accessible buttons help people who use a keyboard, a screen reader, voice control or a switch device, as well as people with low vision or motor impairments.

How to support it

- Use a button for actions and a link for going to a new page. Screen reader users rely on this difference to know what will happen.
- Use the native HTML `<button>` element. It can be reached with the Tab key and works with both Enter and Space, without any extra code.
- Give every button a clear, visible label that describes the action, such as 'Save and continue' rather than 'Click here'.
- If a button only shows an icon, give it an accessible name (for example with `aria-label`) and hide the decorative icon from screen readers.
- Make sure every button has a clear focus state, so keyboard users can see where they are.
- Text on a button needs a colour contrast ratio of at least 4.5:1 (or 3:1 for large text). If the button's edge is what shows it's a button, the edge needs a contrast ratio of at least 3:1 against the background.
- Make buttons big enough to tap easily, with space between them (see Tap Targets).
- Don't rely on colour alone, for example to show that an action is a warning or can't be undone. Make it clear in the button text.
- Avoid disabled buttons where you can. They are often hard to see, and they don't tell the user what they need to do.
- Where you can, reuse a tested button from your design system or component library rather than building a new one.

Useful links

- GOV.UK Design System: Button
- NZ Government Web Accessibility Guide: Buttons for keyboard, mouse and touch
- World Wide Web Consortium (W3C) WAI: Button examples (Accessible Rich Internet Applications (ARIA) Authoring Practices Guide)