

Clickable Video Accessibility & Tracking Checklist

Ship hotspots and overlays that everyone can use and you can measure. The companion article explains every line.

A. Build the layer - hotspots and overlays over the video

- Put hotspots and overlays on a transparent layer above the HTML video element
- Store every position as a % of the frame, never fixed pixels, so it scales on phones
- Time each element with a WebVTT metadata cue; show/hide on the cuechange event
- Decide what each click does: pop-up text, a link, a quiz, or a jump in the video

B. Make it accessible - WCAG 2.2 Level AA (legally required in the EU and US)

- Target at least 24x24 CSS pixels, or enough spacing around it (SC 2.5.8)
- Every hotspot reachable by Tab and activated by Enter/Space, with a visible focus ring (SC 2.1.1)
- Pop-up overlays are dismissible, hoverable, and stay until dismissed (SC 1.4.13)
- Mark hotspots with a shape and a text label - never colour or motion alone

C. Track it - a click is only data if the player reports it

- Emit an xAPI statement on every click (actor - interacted - hotspot), not just at the end
- Use the xAPI Video Profile verbs for the clip itself: played, paused, seeked, completed
- Send statements to a Learning Record Store; with SCORM, log clicks as fixed interactions
- Plan hotspot analytics: which spots get used, which are ignored, where learners drop off

The right-tool test before you build

Use hotspots and overlays for spatial or exploratory learning - labelling equipment, walking through an interface, offering optional depth. They are the cheapest interactivity, so reach for them first. When you need to MEASURE understanding, add an in-player quiz that produces a score. When learners must practise a CONSEQUENTIAL decision, use a branching scenario - far more expensive, but it teaches judgment. And remember: clicking a hotspot is engagement, not mastery; do not let 'clicked all hotspots' count as 'completed'.