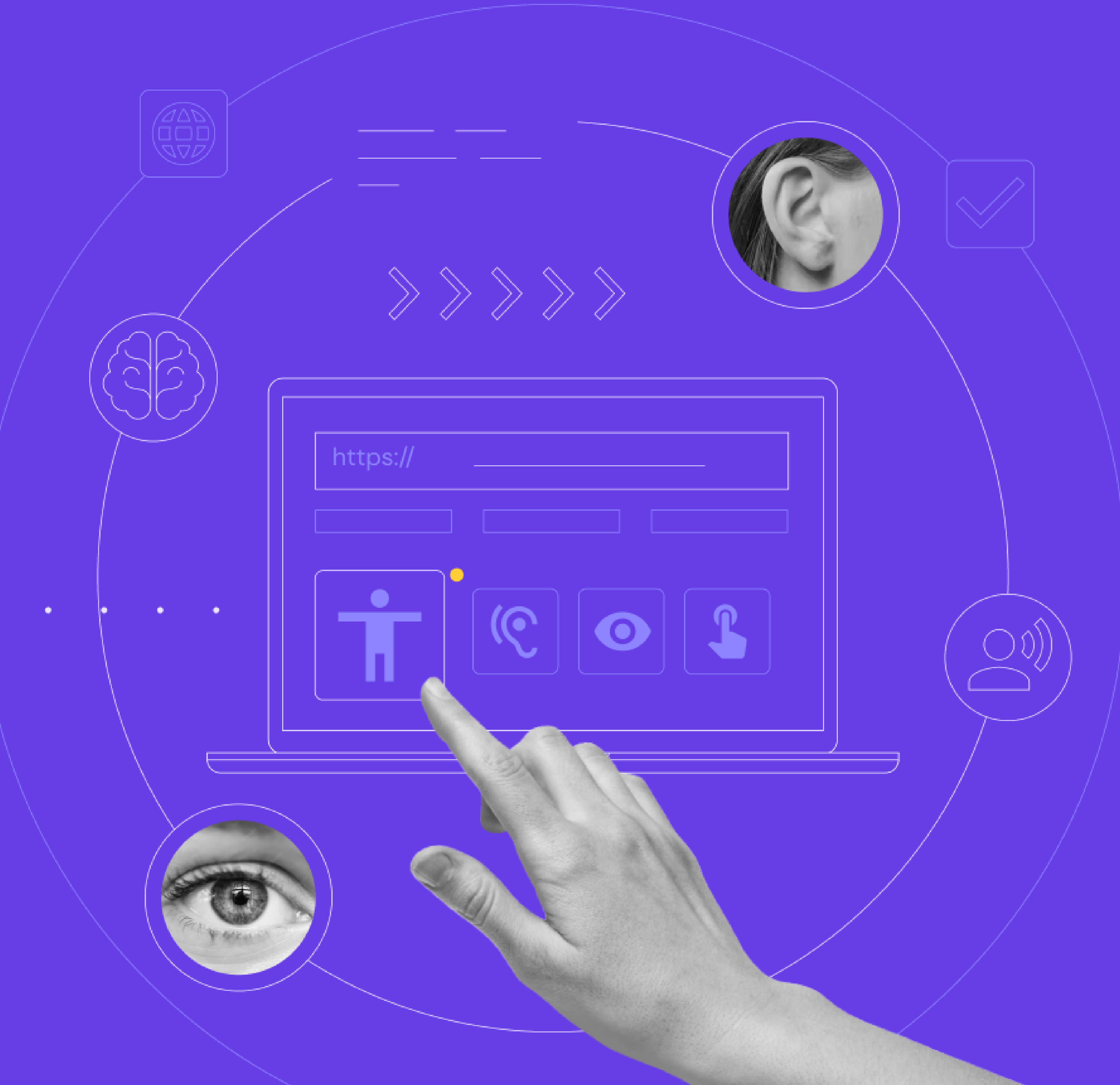
HOSTINGER

Three. Two. Online

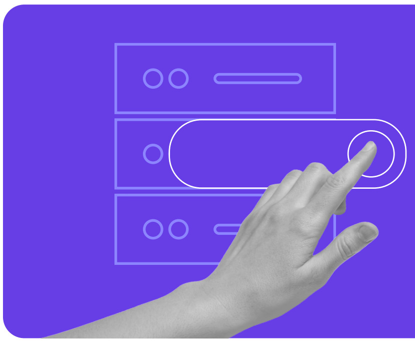
Web Accessibility Checklist



Web Accessibility Checklist

The Web Content Accessibility Guidelines (WCAG) are the fundamental accessibility standardization rules worldwide. By following these guidelines, you can make your website accessible for people with disabilities.

To ensure that your website meets the **WCAG** standards, follow our web accessibility checklist based on the **POUR** principles with bonus tips.



1 Perceivable

Your web content should be easy to perceive for any visitor.

- Users should be able to understand what your visual content is about even if they don't see it.

□ Provide alt text for all images, including icons, buttons, and graphics.

□ Add brief descriptions for video and audio content.

□ Use labels for form input and user interface components, such as search boxes, pagination, and checkboxes.

□ Use null alt text attributes **(alt="")** for decorative images.

- Users should be able to follow your video and audio content easily.

□ Include text transcripts, subtitles, and captions for pre-recorded audio content, such as podcasts and **.mp3** files.

□ Use audio descriptions to describe relevant visual information in a video.

□ Synchronize subtitles to ensure that they appear simultaneously as the audio plays.

□ Avoid using autoplay in media.

- Users should be able to distinguish and see or hear the content easily.

  - □ Combine colors and text cues for form control labels.
  - □ Make sure your text color is easily visible on the background.
  - □ Provide an option to pause, stop, or adjust the audio volume that is played on a website.
  - □ Use a contrast ratio of at least 4:5:1 for text and text images.

- Users should be able to change the content presentation according to their needs without losing information or structure.

  - □ Use semantic mark-up properly for headings, tables, landmarks, and lists.
  - □ Present information and instructions in a logical sequence.
  - □ Do not rely solely on shape, color, size, sound, visual location, or orientation for the instructions.
  - □ Ensure that users can customize the presentation within their browsers and assistive technologies.

2 Operable

The interface elements, such as forms, navigation, and input controls, should be easy to operate with any tool.

- Users should be able to navigate your website using only a keyboard.

  - □ Provide the option to change character-key shortcuts.
  - □ Allow users to disable the single-key shortcuts.
  - □ Ensure that web browsers and authoring tools support keyboard shortcuts.

- Users should have enough time to read the content and complete tasks on a website.

  - □ Give an option to turn off, adjust, or extend the time limit if a page has time limitations.

- □ Allow users to pause, stop, or hide automatically moving, blinking, or scrolling content, such as carousels and animations.
- □ Provide users the opportunity to postpone or suppress notifications, like alerts on the page or system updates.

- Users should not deal with moving, flashing, and blinking content with certain rates and patterns as it can cause seizures or physical reactions.

  - □ Ensure no content flashes more than three times per second on a web page.
  - □ Warn users before flashing content is presented and provide alternatives.
  - □ Provide an option to switch off animations unless they are essential.
  - □ Keep flashing area smaller than 25% of 10 degrees of the visual field, which represents the central area of eye vision or approximately **341 x 256** pixels.

- Users should be able to navigate through the website and find the content easily.

  - □ Create a skip link at the top of each page that goes directly to the main content area.
  - □ Provide descriptive and informative titles for web pages.
  - □ Use a logical and intuitive navigation order for links and form elements, like submit buttons, checkboxes, and radio buttons.

- Users should be able to use your website through various inputs besides a keyboard, such as a voice recognition system and gestures.

  - □ Avoid relying only on multipoint or path-based gestures, such as swiping and pinching the screen.
  - □ Create undo or abort functions to avoid accidental activation.
  - □ Ensure that the active components, like buttons, links, and lists, are large enough to make them easier to activate by touch.

❸ Understandable

The web content and interface components should be explicit and predictable.

- All users should be able to read and understand the content easily.

  □ Identify the primary page language using an HTML lang attribute. For example, **<html lang="en">** for English.

  □ Give definitions for any unfamiliar or ambiguous words through adjacent text or glossary.

  □ Use clear and simple language for all content. For instance, write content for an 8th-grade reading level.

- Web pages should be presented in a predictable way, making them easy to operate.

  □ Have a consistent navigation process on multiple pages.

  □ Use the same labels for user interface elements that are repeated on web pages.

  □ Changes on a web page should be based on user approval.

- Input assistance should help users avoid mistakes when they interact with the content.

  □ Provide descriptive labels for user interface controls, such as text fields, dropdown menus, and list boxes.

  □ Identify required fields that were uncompleted using text descriptions.

  □ Give an example of the expected data format when users need to input data.

④ Robust

Users should be able to use a wide range of agents to access web content, such as assistive technology (AT), mobile browsers, and voice browsers.

- Content should be compatible with the current and future tools.

□ Provide a name, role, and value for all user interface elements.

□ Validate web pages to check whether the markup can work properly across all browsers and AT.

□ Use ARIA to improve accessibility when HTML is not enough.

Bonus Tips

- Use a color contrast checker, like WAVE or TPGI, on your web pages to optimize your content.
- Only use heading 1 for the website's title and page's title so that a screen reader can easily interpret the content.
- Arrange the heading level in a logical way and do not skip it. For example, after using heading 2, do not jump into heading 4.
- Avoid Captcha if possible. Instead, use automatic detections or interface interactions, like checkboxes, radio buttons, or email verification.
- Ensure to expand an acronym on the first use. For instance, assistive technology (AT).
- Use images, illustrations, video, audio, and symbols to help users understand the meaning.
- Avoid using "click here" as an anchor text as it is not descriptive and effective for the user's screen reader. For example, instead of "**Click here** to create an accessible website." write "to create an accessible website, check **our web accessibility checklist**."