

Accessibility in Web Apps

(WCAG Guidelines)

Web accessibility ensures that websites, applications, and tools are usable by people with disabilities. It encompasses a range of practices that enable users with visual, auditory, motor, and cognitive impairments to perceive, understand, navigate, and interact with the web.

The **Web Content Accessibility Guidelines (WCAG)**, developed by the W3C, provide a global standard for accessibility. These guidelines are structured around four core principles: **Perceivable, Operable, Understandable, and Robust (POUR)**.

Practical Accessibility Techniques

Technique	Benefit
Use <code>aria-label / aria-*</code>	Enhances screen reader compatibility
Semantic HTML tags	Better structure for assistive tools
Keyboard focus indicators	Enables keyboard-only navigation
Skip links (<code></code>)	Helps users bypass repeated content
Responsive and zoomable UI	Supports low-vision users
Descriptive link text	Improves clarity for screen readers

Tools for Accessibility Testing

- **Lighthouse (Chrome DevTools)** – Provides accessibility audits.
- **axe DevTools** – Browser extension for detecting WCAG issues.
- **NVDA / VoiceOver / JAWS** – Screen readers for manual testing.
- **Wave (WebAIM)** – Visual feedback on accessibility issues.
- **Keyboard-only testing** – Ensure the app is fully navigable via keyboard.

Example: Accessible Button

```
<!-- Good -->
<button type="submit" aria-label="Submit form">Submit</button>

<!-- Bad -->
<div onclick="submitForm()">Submit</div> <!-- Not accessible -->
```

References

- [WCAG 2.1 Guidelines](#)
- [WebAIM Checklist](#)
- [MDN Accessibility Docs](#)