Introduction to Web Accessibility

Web accessibility means that websites, tools, and technologies are designed and developed so that people with disabilities can use them. More specifically, people can:

- perceive, understand, navigate, and interact with the Web
- contribute to the Web

Accessibility in Context and Impact

Diverse Disabilities

Web accessibility aids those with visual, auditory, motor, cognitive impairments and temporary limitations.

Universal Benefit

Accessible design helps people in noisy environments and on different devices, improving user experience for all.

Defining Web Accessibility and POUR Principles

Accessibility follows POUR: Perceivable, Operable, Understandable, Robust.

01 **Perceivable**

Content is presented for multiple senses, such as sight and sound, to ensure everyone can perceive it.

02 **Operable**

Users can operate the interface, navigation, and controls by various means, like keyboard or voice.

03 **Understandable & Robust**

Content is clear and consistent; it works with a variety of tools like browsers and assistive tech.

Importance for Individuals and Society

Individuals: Equal Access

Ensures everyone, including those with disabilities, can access information and services online.

Businesses: Compliance & Reach

Meets legal standards and expands audience by being more inclusive.

Society: Inclusion

Promotes digital participation and fosters innovation that benefits all users.

Making the Web Accessible: Guidelines

01

WCAG Guidelines

Follow Web Content Accessibility Guidelines set by W3C for global accessibility.

02

Legal Frameworks

Comply with accessibility laws, such as ADA and Section 508 in many countries.

03

Best Practices

Use semantic HTML, provide alt text, ensure color contrast, enable keyboard access, and add captions.

Key Practices for Accessibility

Implement these steps for an accessible site experience for all users.

Semantic HTML

Structure pages with headings, lists, and proper tags to assist screen readers.

Alt Text & Captions

Describe images and provide captions for audio and video content.

Keyboard Access

Ensure users can navigate using only the keyboard.

Evaluating Web Accessibility

Check sites using both manual and automated approaches to accessibility.

01

Manual Keyboard Testing

Use Tab and keyboard shortcuts to test all navigation and interactive elements.

02

Screen Reader Testing

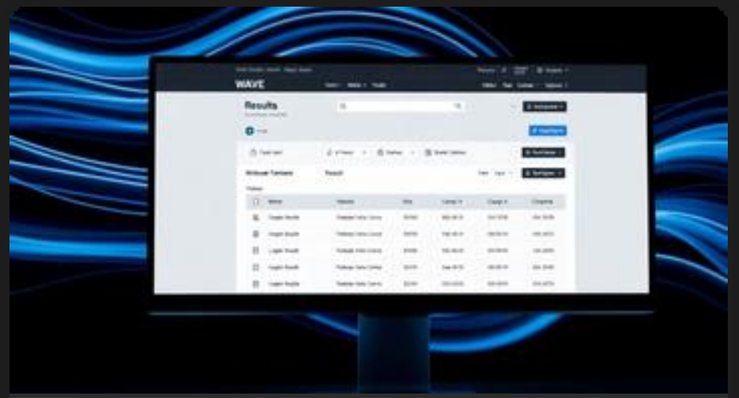
Assess content with tools like **NVDA**, **JAWS**, or **VoiceOver** for reading clarity.

03

Automated Tool Checks

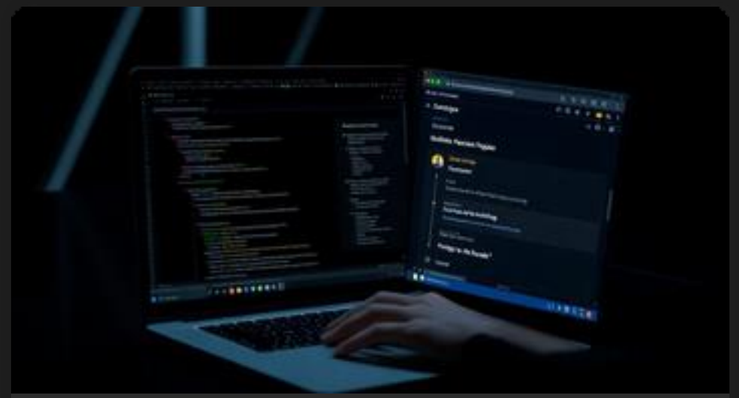
Use tools like **WAVE**, axe **DevTools**, and Lighthouse for rapid accessibility auditing.

Accessibility Testing Tools and Methods



WAVE

Browser extension highlighting issues such as missing alt text, headings, and contrast problems.



axe DevTools

Automates detection of accessibility bugs and suggestions in development workflows.

Lighthouse

Chrome DevTools feature for running accessibility audits and generating improvement reports.

Code Examples for Accessibility

01 **Accessible Image Alt Text**

for descriptive image alternatives.

02 **Semantic and Labeled Forms**

Email: links labels to inputs for screen readers.

03 **Keyboard Navigation Skip Link**

Skip to main content lets keyboard users bypass repetitive links.

04 **Color Contrast in CSS**

body { color: #222; background-color: #fff; } ensures readability for low vision users.