

Corso WCAG 2.1

Rendiamo accessibili i nostri siti
WEB



Scopo del corso



Dopo i corsi su HTML e CSS, **aggiungiamo un tassello fondamentale:**
l'accessibilità web, spesso trascurata ma **essenziale per un design inclusivo.**

Cosa impareremo:

- ✓ Cosa sono le WCAG 2.1 e perché sono importanti
- ✓ Come implementare modifiche concrete in HTML/CSS
- ✓ Strumenti gratuiti per testare l'accessibilità
- ✓ Esempi pratici "prima/dopo" per ogni criterio

Perché questo corso?

- ⚠ Oltre il 15% della popolazione globale ha disabilità (fonte: WHO)
- ⚠ Requisito legale in molti paesi (es. Direttiva UE, ADA negli USA)
- ⚠ Migliora UX e SEO per tutti gli utenti

WCAG – strumenti

Lo strumento che useremo durante il corso è

<https://codepen.io>

CodePen is a social development environment. At its heart, it allows you to write code in the browser, and see the results of it as you build. A useful and liberating online code editor for developers of any skill, and particularly empowering for people learning to code. We focus primarily on front-end languages like HTML, CSS, JavaScript, and preprocessing syntaxes that turn into those things.

Iscrivetevi e seguite il profilo creato apposta per il corso

<https://codepen.io/matteobaccan>

- Codepen.io
- Notepad
- Notepad++
- VisualStudio Code
- Codespace di GitHub

Va bene qualsiasi editor, non visuale, meglio se con syntax highlighter e code completion

<https://github.com/matteobaccan/CORSO-WCAG>

Cosa sono?

Linee guida per rendere i contenuti web accessibili a:

 Non vedenti |  Disabili motori |  Neurodiversi |  Anziani

Chi le sviluppa?

Il **W3C** (World Wide Web Consortium) attraverso un processo:

- ◆ Collaborativo
- ◆ Basato su evidenze
- ◆ Aggiornato periodicamente (WCAG 2.2 in lavorazione)

I 4 principi fondanti (Pour)

1. Perceivable

Informazioni e componenti UI devono essere presentati in modi **percepibili** da tutti i sensi.

Esempi:

- Testo alternativo per immagini
- Sottotitoli per video
- Contrasto colore adeguato

Perché è importante?

Permette a screen reader e dispositivi assistivi di interpretare i contenuti.

I 4 principi fondanti (pOur)

2. Operable

L'interfaccia deve essere **utilizzabile** con diverse modalità di input.

Esempi:

- Navigazione da tastiera
- Tempo sufficiente per leggere i contenuti
- Evitare contenuti lampeggianti

Perché è importante?

Garantisce accesso a utenti con disabilità motorie o epilessia.

I 4 principi fondanti (poUr)

3. Understandable

Contenuti e funzionalità devono essere **chiari e prevedibili**.

Esempi:

- Messaggi d'errore intuitivi
- Linguaggio semplice
- Flusso di navigazione coerente

Perché è importante?

Aiuta utenti con disabilità cognitive o non madrelingua.

I 4 principi fondanti (pouR)

4. Robust

I contenuti devono essere compatibili con **tecnologie attuali e future**.

Esempi:

- HTML semantico validato
- Compatibilità con screen reader
- Utilizzo corretto di ARIA

Perché è importante?

Assicura longevità del sito e adattamento a nuovi dispositivi.

Key Takeaway: I principi POUR sono interdipendenti.

Un sito accessibile li soddisfa *tutti contemporaneamente*.

Evoluzione

Evoluzione WCAG

1999 ➔ WCAG 1.0 | 2008 ➔ WCAG 2.0 | 2018 ➔ WCAG 2.1 | 2023 ➔ WCAG 2.2

Livelli di Conformità

- **A:** Requisiti base (obbligatorio per legge in UE)
- **AA:** Standard raccomandato (soddisfa la maggior parte delle disabilità)
- **AAA:** Ottimizzazione avanzata (contesti specifici)

Perceivable

Le informazioni e i componenti dell'interfaccia utente devono essere presentabili agli utenti in modo che possano percepirla.

- **1.1 Text Alternatives (A)**: Testo alternativo per contenuti non testuali
- **1.2 Time-based Media (A)**: Contenuti multimediali con alternative testuali
- **1.3 Adaptable (A)**: Contenuti strutturati in modo semantico
- **1.4 Distinguishable (A)**: Facilità di percezione dei contenuti e delle UI

1.1 Text Alternatives (A)

I contenuti non testuali devono avere un'alternativa testuale che consenta agli utenti di comprendere il contenuto.

1.1 Testo Alternativo (A)

Prima

```

```

Problema: Mancanza di alt text per non vedenti.

Dopo

```

```

Strumenti:

- WAVE <https://wave.webaim.org/>
- Lighthouse <https://developer.chrome.com/docs/lighthouse>

1.2 Time-based Media

Ogni contenuto multimediale deve avere un'alternativa testuale.

- **1.2.1** Audio-only e Video-only (A): Trascrizione o audio descrittivo
- **1.2.2** Captions (A): Sottotitoli per video
- **1.2.3** Audio Description or Media Alternative (A): Descrizione audio o alternativa testuale
- **1.2.4** Captions (Live) (AA): Sottotitoli in tempo reale
- **1.2.5** Audio Description (Prerecorded) (AA): Descrizione audio per video preregistrati
- **1.2.6** Sign Language (Prerecorded) (AAA): Lingua dei segni per video preregistrati
- **1.2.7** Extended Audio Description (AAA): Descrizione audio estesa per video
- **1.2.8** Media Alternative (AAA): Alternativa testuale per media
- **1.2.9** Audio-only (AAA): Trascrizione per audio-only

1.2.1 Audio-only e Video-only (A)

Prima

```
<audio controls>
  <source src="audio.mp3" type="audio/mpeg">
</audio>
```

Problema: Non è fornita una trascrizione per il contenuto audio, rendendolo inaccessibile agli utenti non udenti.

1.2.1 Audio-only e Video-only (A) soluzione

Dopo

```
<audio controls>
  <source src="audio.mp3" type="audio/mpeg">
</audio>
<p>Trascrizione: "Benvenuti al nostro podcast. Oggi parleremo di accessibilità web e delle WCAG 2.1..."</p>
```

Soluzione: Aggiunta di una trascrizione testuale per il contenuto audio.

Strumenti:

- [Otter.ai](#) per generare trascrizioni.
- [Sonix](#) per trascrizioni automatiche.

1.2.2 Captions (A)

Prima

```
<video controls>
  <source src="video.mp4">
</video>
```

Problema: Mancano sottotitoli.

Dopo

```
<video controls>
  <source src="video.mp4">
  <track src="sottotitoli.vtt" kind="captions" label="Italiano">
</video>
```

Strumenti:

- Amara (per creare sottotitoli) - YouTube Studio (auto-captioning)

1.2.3 Audio Description or Media Alternative (A) - prima

Prima

```
<video controls>
  <source src="video.mp4" type="video/mp4">
</video>
```

Problema: Il video non include una descrizione audio o un'alternativa testuale, rendendolo inaccessibile agli utenti non vedenti che non possono percepire i contenuti visivi.

1.2.3 Audio Description or Media Alternative (A) - Dopo

Dopo

```
<video controls>
  <source src="video.mp4" type="video/mp4">
  <track src="audio-descrizione.vtt" kind="descriptions" label="Descrizione Audio">
</video>
<p>Alternativa Testuale: "In questo video, una persona dimostra come utilizzare uno screen reader per navigare un sito web."</p>
```

1.2.3 Audio Description or Media Alternative (A) - soluzione

Soluzione:

- Aggiunta di una traccia di descrizione audio (`kind="descriptions"`) per descrivere i contenuti visivi.
- Fornita un'alternativa testuale come opzione di fallback per gli utenti che non possono accedere alla descrizione audio.

Strumenti:

- [Amara](#) per creare descrizioni audio.
- [YouDescribe](#) per generare e condividere descrizioni audio.

1.2.4 Captions (Live) (AA)

Prima

```
<video controls>
  <source src="live-stream.mp4" type="video/mp4">
</video>
```

Problema: Il video in diretta non include sottotitoli in tempo reale, rendendolo inaccessibile agli utenti non udenti o con difficoltà uditive.

1.2.4 Captions (Live) (AA) - dopo

Dopo

```
<video controls>
  <source src="live-stream.mp4" type="video/mp4">
    <track kind="captions" src="live-captions.vtt" srclang="it" label="Italiano">
</video>
```

Soluzione:

- Aggiunta di sottotitoli in tempo reale per il video in diretta.
- Utilizzo di un file `.vtt` aggiornato dinamicamente per fornire i sottotitoli.

Strumenti:

- [OBS Studio](#) con plugin per sottotitoli live.
- [YouTube Live Captioning](#) per generare sottotitoli automatici.
- [Web Captioner](#) per trascrizioni in tempo reale.

Altre specifiche 1.2

1.2.5 Audio Description (Prerecorded) (AA)

1.2.6 Sign Language (Prerecorded) (AAA)

1.2.7 Extended Audio Description (AAA)

1.2.8 Media Alternative (AAA)

1.2.9 Audio-only (AAA)

1.3 Adaptable

- **1.3.1 Info and Relationships (A)**: Struttura semantica dei contenuti
- **1.3.2 Meaningful Sequence (A)**: Sequenza significativa dei contenuti
- **1.3.3 Sensory Characteristics (A)**: Caratteristiche sensoriali per identificare elementi
- **1.3.4 Orientation (AA)**: Orientamento del layout
- **1.3.5 Identify Input Purpose (AA)**: Identificazione del tipo di input
- **1.3.6 Identify Purpose (AAA)**: Identificazione del tipo di contenuto

1.3.1 Tabelle Accessibili

Prima

```
<div>Nome</div><div>Età</div>
<div>Mario</div><div>30</div>
```

Problema: Struttura non semantica per dati tabellari.

Dopo

```
<table>
  <caption>Dipendenti</caption>
  <thead><tr><th scope="col">Nome</th><th scope="col">Età</th></tr></thead>
  <tbody><tr><td>Mario</td><td>30</td></tr></tbody>
</table>
```

Strumenti:

- Screen reader test con NVDA - Table Inspector in WAVE

1.3.2 Meaningful Sequence (A)

Prima

```
<div>
  <h2>Benvenuto</h2>
  <p>Questo è un corso sulle WCAG.</p>
  <h1>Corso WCAG</h1>
</div>
```

Problema: La sequenza del contenuto non è significativa. Gli screen reader leggeranno il titolo principale dopo il sottotitolo, causando confusione.

1.3.2 Meaningful Sequence (A) - Dopo

Dopo

```
<div>
  <h1>Corso WCAG</h1>
  <h2>Benvenuto</h2>
  <p>Questo è un corso sulle WCAG.</p>
</div>
```

Soluzione:

- Organizzare il contenuto in una sequenza logica e significativa per garantire che gli screen reader leggano le informazioni nell'ordine corretto.

Strumenti:

- [axe DevTools](#) per verificare la struttura del contenuto.
- Screen reader (es. NVDA o VoiceOver) per testare l'ordine di lettura.

1.3.3 Sensory Characteristics (A)

Prima

```
<p>Premi il pulsante rosso per continuare.</p>
<button style="background-color: red;">Continua</button>
```

Problema: L'istruzione si basa esclusivamente su caratteristiche sensoriali (colore), rendendo difficile l'interazione per utenti con disabilità visive o daltonismo.

1.3.3 Sensory Characteristics (A) - Dopo

Dopo

```
<p>Premi il pulsante con l'etichetta "Continua" per proseguire.</p>
<button style="background-color: red;" aria-label="Continua">Continua</button>
```

Soluzione:

- Fornire istruzioni che non si basino esclusivamente su caratteristiche sensoriali come colore, forma o posizione.
- Aggiungere un'etichetta accessibile (`aria-label`) per descrivere il pulsante.

Strumenti:

- [Contrast Checker](#) per verificare il contrasto.
- Screen reader (es. NVDA o VoiceOver) per testare l'accessibilità.

1.3.3 Sensory Characteristics (A)

Prima

```
<p>Premi il pulsante rosso per continuare.</p>
<button style="background-color: red;">Continua</button>
```

Problema: L'istruzione si basa esclusivamente su caratteristiche sensoriali (colore), rendendo difficile l'interazione per utenti con disabilità visive o daltonismo.

1.3.3 Sensory Characteristics (A) - Dopo

Dopo

```
<p>Premi il pulsante con l'etichetta "Continua" per proseguire.</p>
<button style="background-color: red;" aria-label="Continua">Continua</button>
```

Soluzione:

- Fornire istruzioni che non si basino esclusivamente su caratteristiche sensoriali come colore, forma o posizione.
- Aggiungere un'etichetta accessibile (`aria-label`) per descrivere il pulsante.

Strumenti:

- [Contrast Checker](#) per verificare il contrasto.
- Screen reader (es. NVDA o VoiceOver) per testare l'accessibilità.

Distinguishable 1.4

Rendere facile agli utenti percepire i contenuti e le UI.

- **1.4.1 Use of Color (A)**: Non usare solo il colore per trasmettere informazioni
- **1.4.2 Audio Control (A)**: Controllo audio per contenuti audio
- **1.4.3 Contrast (Minimum) (AA)**: Contrasto minimo 4.5:1 per testo normale
- **1.4.4 Resize Text (AA)**: Testo ridimensionabile fino al 200%
- **1.4.5 Images of Text (AA)**: Testo non come immagine
- **1.4.6 Contrast (Enhanced) (AAA)**: Contrasto minimo 7:1 per testo normale
- **1.4.7 Low or No Background Audio (AAA)**: Audio di sottofondo basso o assente

Distinguishable 1.4 - continua

- **1.4.8** Visual Presentation (AAA): Presentazione visiva dei contenuti
- **1.4.9** Images of Text (No Exception) (AAA): Immagini di testo non ammesse
- **1.4.10** Reflow (AA): Contenuti reflowabili senza perdita di informazioni
- **1.4.11** Non-text Contrast (AA): Contrasto non testuale 3:1
- **1.4.12** Text Spacing (AA): Spaziatura del testo
- **1.4.13** Content on Hover or Focus (AA): Contenuti visibili al passaggio del mouse o al focus

1.4.1 Use of Color (A)

Prima

```
<p>Premi il pulsante verde per confermare e il pulsante rosso per annullare.</p>
<button style="background-color: green;">Conferma</button>
<button style="background-color: red;">Annulla</button>
```

Problema: Le istruzioni si basano esclusivamente sul colore, rendendo difficile l'interazione per utenti con disabilità visive o daltonismo.

1.4.1 Use of Color (A) - Dopo

Dopo

```
<p>Premi il pulsante con l'etichetta "Conferma" per confermare e il  
pulsante con l'etichetta "Annulla" per annullare.</p>  
<button style="background-color: green;" aria-label="Conferma">Conferma</button>  
<button style="background-color: red;" aria-label="Annulla">Annulla</button>
```

Soluzione:

- Fornire istruzioni che non si basino esclusivamente sul colore.
- Aggiungere etichette accessibili (`aria-label`) per descrivere i pulsanti.

Strumenti:

- [Contrast Checker](#) per verificare il contrasto.
- Test con screen reader (es. NVDA o VoiceOver).

1.4.3 Contrast (Minimum) (AA)

Prima

```
.button { background: #888; color: #fff; } /* Contrasto 4:1 */
```

Problema: Contrasto insufficiente (< 4.5:1 per testo normale).

Dopo

```
.button { background: #0056b3; color: #fff; } /* Contrasto 7:1 */
```

Strumenti:

- Contrast Checker <https://webaim.org/resources/contrastchecker/>
- Color Contrast: <https://dequeuniversity.com/rules/axe/4.10/color-contrast>

Operable 2.1

Le interfacce utente e la navigazione devono essere operabili.

- **2.1 Keyboard Accessible (A)**: Tutte le funzionalità devono essere accessibili da tastiera
- **2.2 Enough Time (A)**: Tempo sufficiente per completare le attività
- **2.3 No Timing (A)**: Nessun limite di tempo per completare le attività
- **2.4 Navigable (A)**: Navigazione e ricerca facili
- **2.5 Input Modalities (A)**: Supporto per diverse modalità di input

2.1.1 Keyboard Accessible (A)

Prima

```
<div class="dropdown" onclick="openMenu()">Menu</div>
```

Problema: Non navigabile via tastiera (manca `tabindex`).

Dopo

```
<div class="dropdown" tabindex="0" role="button" aria-expanded="false">Menu</div>
```

Verifica:

- Usa `Tab` e `Enter` per testare
- Strumento: Total1y (khan.github.io/tota11y/)

2.3.3 Animation from Interactions (AAA)

Prima

```
.slider { transition: transform 0.8s; }
```

Problema: Animazioni potenzialmente dannose per utenti sensibili.

Dopo

```
@media (prefers-reduced-motion: reduce) {  
    .slider { transition: none; }  
}
```

Strumenti:

- Simulatore preferenze movimento: Chrome DevTools > Rendering
- CSS Media Queries Level 5 validator

2.4.1 Bypass Blocks (A)

Prima

```
<header>
  <nav>...</nav>
</header>
```

Problema: Nessun modo per saltare la navigazione.

2.4.1 Bypass Blocks (A) - Dopo

Dopo

```
<a href="#main" class="skip-link">Salta al contenuto</a>
<nav>...</nav>
<main id="main">...</main>

<style>
.skip-link {
    position: absolute;
    left: -9999px;
}
.skip-link:focus {
    left: 10px;
}
</style>
```

Strumenti:

- Test con screen reader + tastiera - HeadingMap estensione

2.4.4 Link Purpose (In Context) (A)

Prima

```
<a href="/blog">Clicca qui</a>
```

Problema: Testo non contestuale fuori contesto.

Dopo

```
<a href="/blog">Leggi gli ultimi articoli del blog</a>
```

Verifica:

- Lista link con Web Developer Toolbar
- Strumento: Link Text Analyzer (<https://axesslab.com/link-texts/>)

2.4.7 Focus Visible (AA)

Prima

```
.button:focus { outline: none; }
```

Problema: Focus non visibile per utenti keyboard-only.

Dopo

```
.button:focus {  
    outline: 3px solid #0056b3;  
    outline-offset: 2px;  
}
```

Strumenti:

- Tasti `Tab` + `Shift+Tab` per test manuale
- Focus Order Viewer in axe DevTools

Understandable

I contenuti e le operazioni dell'interfaccia utente devono essere comprensibili.

- **3.1 Readable (A)**: Testo leggibile e comprensibile
- **3.2 Predictable (A)**: Interfaccia prevedibile e coerente
- **3.3 Input Assistance (A)**: Aiuto per l'input e la correzione degli errori

3.1.1 Language of Page (A)

Prima

```
<html>
<head><title>Welcome</title></head>
```

Problema: Lingua non dichiarata per screen reader.

Dopo

```
<html lang="it">
<head><title>Benvenuto</title></head>
```

Verifica:

- Validatore HTML (<https://validator.w3.org/>)
- Strumento: axe "html-has-lang" rule

3.3.2 Labels or Instructions (A)

Prima

```
<input type="text" name="email">
```

Problema: Manca label associata.

Dopo

```
<label for="email">Email:</label>
<input type="email" id="email" aria-describedby="email-help">
<span id="email-help" class="sr-only">Inserisci un indirizzo valido</span>
```

CSS per screen reader:

```
.sr-only { position: absolute; left: -10000px; }
```

Robust

I contenuti devono essere robusti e compatibili con le tecnologie assistive.

- 4.1 Compatible

4.1.1 Parsing (A)

Prima

```
<div>
  <p>Benvenuto</p>
  <span>Corso WCAG</span>
</div>
```

Problema:

- Struttura HTML non valida o mal formattata.
- Tag non chiusi correttamente o annidati in modo errato, causando problemi con i browser e le tecnologie assistive.

4.1.1 Parsing (A) - Dopo

Dopo

```
<div>
  <p>Benvenuto</p>
  <p>Corso WCAG</p>
</div>
```

Soluzione:

- Correggere la struttura HTML per garantire che sia valida e ben formattata.
- Utilizzare tag semanticci e chiuderli correttamente.

Strumenti:

- [W3C HTML Validator](#) per verificare la validità del codice.
- [axe DevTools](#) per controllare errori di parsing.

4.1.2 Name, Role, Value (A)

Prima

```
<div>Scopri di più</div>
```

Problema: Elemento non semantico per bottoni/navigazione.

Dopo

```
<button aria-label="Apri menu">Scopri di più</button>
<nav aria-label="Menu principale">...</nav>
```

Strumenti:

- HTML Validator (validator.w3.org)
- Axe DevTools (estensione browser)
- IBM Accessibility Equal Access Toolkit: Accessibility Checker

4.1.3 Status Messages (AA)

Prima

```
<div id="status">Il tuo ordine è stato inviato!</div>
```

Problema: Gli screen reader non notificano automaticamente i cambiamenti di stato, rendendo difficile per gli utenti con disabilità visive percepire aggiornamenti importanti.

Dopo

```
<div id="status" role="status" aria-live="polite">Il tuo ordine è stato inviato!</div>
```

Soluzione:

- Aggiungere l'attributo `role="status"` per indicare che si tratta di un messaggio di stato.
- Utilizzare `aria-live="polite"` per notificare i cambiamenti senza interrompere l'utente.

Nota:

- Usare `aria-live="assertive"` per messaggi critici che richiedono attenzione immediata.

Fonti

- **WCAG 2.1 Full Guide:** <https://www.w3.org/TR/WCAG21/>
- **WebAIM Checklist:** <https://webaim.org/standards/wcag/checklist>
- **MDN Accessibility Guide:** <https://developer.mozilla.org/en-US/docs/Web/Accessibility>
- **Definizioni e argomenti:** <https://it.wikipedia.org>

Ogni immagine inserita riporta la fonte

Disclaimer

Questo materiale è stato realizzato con le seguenti modalità:

- Contenuto testuale

Redatto attraverso sistemi di AI per la generazione della bozza iniziale, successivamente rielaborato, verificato e integrato manualmente dall'autore.

- Elementi grafici

L'immagine di sfondo è stata generata tramite [Haikei.app](#). Eventuali altri elementi visivi derivano da banche immagini royalty-free o creazioni originali.

- Ricerche

I dati e le informazioni citate sono state raccolte da fonti pubbliche accessibili online, selezionate e contestualizzate in modo critico dall'autore.

La direzione intellettuale, le scelte contenutistiche e l'accuratezza delle informazioni restano sotto la piena responsabilità dell'autore.