

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>

WCAG – editor





- 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/CorsoWCAG>

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. Percepibile

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. Operabile

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. Uncomprensibile

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. Robusto

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)

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

- **1.1** Non-text Content (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

1.1.1 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)

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)

1.2.4 Captions (Live) (AA)

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

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
- **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

Contrasto Colori (1.4.3) (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>

Responsive Design (1.4.10)

Prima

```
<meta name="viewport" content="width=device-width">
```

CSS problematico:

```
.container { width: 1200px; }
```

Dopo

```
.container { max-width: 100%; }  
@media (max-width: 768px) { ... }
```

Strumenti:

- Browser DevTools (responsive mode) - Lighthouse (mobile accessibility)

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

Navigazione da Tastiera (2.1.1) (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: Tota11y (khan.github.io/tota11y/)

Animazioni e Motion (2.3.3)

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

Saltare Contenuti Ripetitivi (2.4.1)

Prima

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

Problema: Nessun modo per saltare la navigazione.

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 {
```

Link Descrittivi (2.4.4)

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/>)

Focus Visibility (2.4.7)

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

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

10. Lingua della Pagina (3.1.1)

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

Form Accessibili (3.3.2) (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; }
```

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

HTML Semantico (4.1.2) (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

Ruoli ARIA per Widget (4.1.2)

Prima

```
<div class="modal">...</div>
```

Problema: Ruolo non definito per componenti complessi.

Dopo

```
<div class="modal" role="dialog" aria-labelledby="modalTitle">  
  <h2 id="modalTitle">Conferma ordine</h2>  
  ...  
</div>
```

Strumenti:

- ARC Toolkit (<https://www.paciellogroup.com/toolkit/>)
- Accessible Name Calculator

ARIA per Dinamicità (4.1.3)

Prima

```
<div id="alert">Nuovo messaggio!</div>
```

Problema: Screen reader non notificano cambiamenti.

Dopo

```
<div id="alert" role="alert" aria-live="assertive">Nuovo messaggio!</div>
```

Verifica:

- NVDA Screen Reader ([nvaccess.org](https://www.nvaccess.org))
- VoiceOver (macOS)
- Simulatore Daltonismo <<https://www.toptal.com/designers/colorfilter>

Testing Completo

Checklist:

1. Validazione HTML/CSS
2. Test contrasto colori
3. Navigazione solo da tastiera
4. Screen reader test (es. JAWS, NVDA)
5. Lighthouse Audit

Tool Gratuiti:

- axe DevTools
- WAVE Evaluation Tool
- Pa11y (pa11y.org)

- **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.