

Tecnologie Web

Matteo Mazzaretto

2025/2026

Indice

1 ID e Class	1
2 Procedimento a cascata	1
3 Significato di PURO (4 principi fondamentali dell'accessibilità in un sito web)	1
4 Doppia A o tripla A per l'accessibilità?	2
5 Indicatori di specificità	2
6 Rapporto fra meta tag description e la SERP, definizione SEO	2
7 Globalizzazione vs localizzazione	3
8 Descrivere i bisogni degli utenti con il metodo della pesca	3
9 Struttura ottimale gerarchia	4
10 Area sicura per la visualizzazione	4
11 Ampiezza e profondità per un sito web	4
12 Emotional design	4
13 Differenze fra metodo GET e POST	5
14 Come costruire una tabella accessibile	6
15 Schema o struttura organizzativa	7
15.1 Sequenza	8
15.2 Gerarchia	8
15.3 Faccette o Matrici	8
15.4 Ipertesto	8
16 Si può capire con solo test automatici se un sito è accessibile	8
17 Distinzione fra presentazione, struttura e comportamento	9

18 Nascondere, anche solo parzialmente, il contenuto dei menù tramite menù a tendina è corretto?	9
19 Differenza tra convenzioni interne ed esterne	9
20 Mettere in ordine di applicazione i seguenti stili CSS	9
21 Qual è il media più accessibile?	10
22 Quali sono le tra domande più importanti alle quali devo saper rispondere, pena il fenomeno del disorientamento?	10

1 ID e Class

id e class sono attributi HTML utilizzati come selettori nel CSS.

La class definisce un gruppo di appartenenza, può essere assegnata a più elementi HTML e lo stesso elemento può avere più classi.

Nel CSS si seleziona con `.nomeClasse{ }`, dove tra le parentesi sono definiti gli attributi CSS con i relativi valori

L'id identifica un elemento in modo univoco all'interno del documento HTML.
I principali utilizzi sono:

- selettore in un foglio di stile
- riferimento all'interno di script
- ancora di destinazione di un link
- identificatore generico per il trattamento dei dati

Un id deve essere unico nel documento e deve iniziare con una lettera o con il carattere di sottolineatura (-).

Nel CSS si seleziona con `#nomeId{ }`, dove tra le parentesi sono definiti gli attributi CSS con i relativi valori.

2 Procedimento a cascata

Il “procedimento a cascata” definisce le priorità con cui le regole CSS vengono applicate:

1. Impostazioni personali dell'utente
2. Impostazioni di stile inline definite dall'autore della pagina
3. Fogli di stile embedded definiti dall'autore
4. Fogli di stile esterni definiti dall'autore
5. Impostazioni di stile predefinite del browser
6. Impostazioni utilizzate quando qualcosa non è definito o se il browser non supporta i CSS

Le impostazioni di default cambiano da browser a browser

3 Significato di PURO (4 principi fondamentali dell'accessibilità in un sito web)

PURO vuol dire ”Percepibile, Comprensibile, Robusto, Utilizzabile”.

Se qualcosa è PURO allora è anche accessibile, serve ad avere un metro di riferimento per capire se si è raggiunta l'accessibilità

4 Doppia A o tripla A per l'accessibilità?

Il livello di priorità è determinato dall'impatto che ciascun punto di controllo ha sull'accessibilità:

- Priorità 1 (A): lo sviluppatore deve conformarsi al presente punto di controllo, pena la preclusione di una o più categorie di utenti dall'accesso alle informazioni. Costituisce un requisito base.
- Priorità 2 (AA): lo sviluppatore dovrebbe conformarsi al presente punto di controllo, altrimenti ad una o più categorie diutenti risulterà difficile accedere alle informazioni. Consente di rimuovere barriere significative.
- Priorità 3 (AAA): lo sviluppatore può tenere in considerazione questo punto di controllo, altrimenti ad una o più categorie di utenti sarà in qualche modo ostacolata nell'accedere alle informazioni. Migliora l'accesso ai documenti web.

Per l'accessibilità è necessario rispettare tutti i requisiti dei livelli 1 (A) e 2 (AA).

Il livello 3 (AAA) è opzionale e dedicato a categorie specifiche di utenti; applicarlo in modo rigoroso potrebbe limitare l'usabilità per altri utenti.

5 Indicatori di specificità

Un figlio eredita le impostazioni del padre.

Con più regole a stessa importanza viene applicata l'ultima definita.

L'importanza si calcola con gli indicatori di specificità.

La regola più specifica sarà quella di cui viene applicata la regola.

Date le regole, si calcolano così:

(num !important, num id, num attributi, num tag html).

È molto importante la virgola fra un indicatore e l'altro per permettere la divisione degli elementi.

Esempio:

```
1 #nav a{  
2     color:orange;  
3 }  
4 a{  
5     color:blue;  
6 }
```

La specificità per il primo è (0, 1, 0, 1).

La specificità per il secondo è (0, 0, 0, 1).

Quindi viene applicata la prima regola in caso di contrasti.

6 Rapporto fra meta tag description e la SERP, definizione SEO

Ogni pagina web contiene diversi meta tag, tra cui `title`, `keywords` e `description`:

- **Title:** deve essere dal particolare al generale, con un massimo di circa 60 caratteri. Rappresenta il titolo visualizzato nei risultati dei motori di ricerca.
- **Keywords:** deve contenere parole chiave presenti nella pagina; la maggior parte dovrebbe comparire anche nelle intestazioni per migliorare il ranking.
- **Description:** non influisce direttamente sul ranking, ma fornisce la descrizione visualizzata nei risultati di ricerca. Deve riassumere accuratamente il contenuto della pagina, evitando clickbait, e aiutare l'utente a decidere se aprire il sito.

La SERP (Search Engine Response Page) è la pagina di risposta che si visualizza dopo una ricerca in un motore di ricerca.

La SEO (Search Engine Optimization) indica tutte le attività di ottimizzazione di un sito web volte a migliorarne il posizionamento nelle pagine dei risultati dei motori di ricerca.

7 Globalizzazione vs localizzazione

Nella progettazione di un sito web è fondamentale identificare l'audience principale.

Occorre stabilire se il sito è destinato a una singola nazione o a più culture. Le opzioni principali sono:

- Sito nazionale: accessibile solo da una specifica nazione.
- Sito globale uniforme: stesso contenuto in tutti i paesi (es. GitHub).
- Siti localizzati: versioni diverse del sito per ciascun paese o cultura (es. siti di automobili).

8 Descrivere i bisogni degli utenti con il metodo della pesca

Gli utenti hanno obiettivi diversi quando navigano in un sito. La metafora della pesca identifica i principali comportamenti di navigazione:

- **Tiro perfetto:** l'utente sa esattamente cosa cercare.
- **Trappola per aragoste:** l'utente non ha un obiettivo definito e impara durante la navigazione per iterazioni successive.
- **Pesca con la rete:** l'utente vuole esplorare diversi elementi relativi a un argomento.
- **Boa di segnalazione:** l'utente vuole ritrovare un elemento informativo già identificato come utile.

9 Struttura ottimale gerarchia

La struttura gerarchica è la base per una buona struttura informativa, rende gli utenti in grado di sviluppare facilmente un modello mentale della struttura del sito e la loro localizzazione in questa struttura.

Deve rispettare i requisiti indicati nella domanda [11](#)

10 Area sicura per la visualizzazione

L'area sicura per la visualizzazione è la porzione di pagina web visibile immediatamente senza bisogno di scorrere.

Deve contenere le informazioni principali e di primo impatto, tra cui:

- Elementi informativi fondamentali
- Elementi essenziali per l'interazione
- Elementi grafici per l'identità del sito

11 Ampiezza e profondità per un sito web

Nella progettazione di una struttura gerarchica è necessario trovare un buon equilibrio tra ampiezza (numero di opzioni ad ogni livello) e profondità della gerarchia (numero di livelli):

- gerarchie troppo ampie portano al sovraccarico cognitivo: per il web è consigliabile non andare oltre le 10 opzioni nel menù principale
- gerarchie troppo profonde rendono eccessivo il numero di click necessari per reperire l'informazione: è buona regola non superare i 4 o 5 livelli

I siti in evoluzione possono richiedere una riprogettazione della gerarchia, per questo gerarchie ampie e poco profonde sono facilmente aggiornabili.

Se è possibile organizzare un menù secondo un ordine conveniente per l'utente, le strutture ampie sono preferibili a quelle profonde.

12 Emotional design

L' emotional design si basa sul fatto che le esperienze legate alle emozioni hanno un forte impatto nella memoria a lungo termine e sono richiamate con maggiore accuratezza.

Un design deve tenere conto delle emozioni per trasformare un utente casuale in un accanito lettore/cliente.

Uno dei modi è l'utilizzo del **contrasto**, ovvero l'emozione che rappresenta ciò che l'utente non si aspetta, l'interruzione di un pattern conosciuto.

Il contrasto può essere:

- visuale (differenze nella forma, aspetto, colori)
- cognitivo (ricordi)

Altre tecniche sono:

- l'umanizzazione del prodotto
- l'utilizzo di una personalità
- l'utilizzo delle emozioni come:
 - sorpresa
 - piacere
 - anticipazione
 - status/esclusività
 - rewards

13 Differenze fra metodo GET e POST

Metodo GET: è il predefinito, serve a recuperare informazioni e passa le stringhe in chiaro.

Il browser allega la stringa di query all'url:

- `http://server/path/file.php?parametro=valore`
- limite alla lunghezza della stringa (256 caratteri)
- vulnerabilità dell'accesso

Metodo POST: la stringa di query viene passato come input standard, il che vuol dire maggiore facilità di gestione.

Serve a passare stringhe importanti che non possono essere catturate direttamente, come le password. In PHP i parametri vengono memorizzati in variabili superglobali:

- Con GET: i dati sono accessibili tramite `$_GET`
- Con POST: i dati sono accessibili tramite `$_POST`
- Indipendentemente dal metodo, i dati sono sempre presenti in `$_REQUEST`, permettendo allo script di non dover distinguere il metodo di invio

`$_REQUEST` è indipendente da `$_GET` e `$_POST`: modificarne i valori non influisce sulle altre variabili e viceversa.

Il metodo che serve a passare file è **POST**.

Il metodo POST consente l'invio di dati binari tramite form con l'attributo `enctype="multipart/form-data"`, necessario per l'upload di file. GET non supporta l'invio di file perché i parametri vengono passati nell'URL.

14 Come costruire una tabella accessibile

2. Tabella Accessibile

Scrivere il codice HTML5 che permette di visualizzare a video questa tabella. La tabella deve essere accessibile.

Medagliere Olimpiadi Tokyo 2021

Italia			
Medaglie d'Oro	Medaglie d'Argento	Medaglie di Bronzo	Totali
10	10	20	40
Atleta	Medaglie	Specialità	
Marcell Jacobs	2 medaglie d'Oro	100m maschile, staffetta 4x100m	
Vanessa Ferrari	1 medaglia d'argento	Ginnastica Artistica, corpo libero	
Gianmarco Tamberi	1 medaglia d'oro	Salto in alto maschile	

Per prima cosa bisogna inserire un paragrafo con il riassunto della tabella. La tabella deve avere aria-describedby="sum" che indica il paragrafo di riassunto precedente.

Successivamente ci deve essere la caption, inoltre ogni th deve avere uno scope che indica se è colonna singola ("col") oppure colonna multipla ("colgroup"). Si inserisce scope="row" per indicare la riga singola. Da ricordare che th è utilizzato per le intestazioni. Se possibile inserire le abbreviazioni con abbr assieme agli scope.

Ricordare che <tr> indica la riga mentre <td> indica la colonna.

```

1   <p id="sum">Tabella organizzata per colonne con il numero di
2     medaglie d'oro, d'argento, di bronzo e totali vinte dall'
3     Italia. La seconda parte della tabella contiene il nome dell'
4     atleta, le medaglie vinte e la specialita.
5   </p>
6   <table aria-describedby="sum">
7     <caption>Medagliere Olimpiadi Tokyo 2021</caption>
8     <thead>
9       <tr>
10      <th scope="colgroup" colspan="4">Italia</th>
11    </tr>
12    <tr>
13      <th scope="col" abbr="Oro"> Medaglie d'Oro</th>
14      <th scope="col" abbr="Argento">Medaglie d'Argento</th>
15      <th scope="col" abbr="Bronzo">Medaglie di Bronzo</th>
16      <th scope="col" abbr="Tot">Totali</th>
17    </tr>
18  </thead>
19  <tbody>
20    <tr>
21      <td>10</td>
22      <td>10</td>
23      <td>20</td>
24      <td>40</td>
25    </tr>
26    <tr>
27      <th scope="col">Atleta</th>
28      <th scope="col">Medaglie</th>
29      <th scope="colgroup" colspan="2">Specialita</th>
30    </tr>
31    <tr>
32      <th scope="row" lang="en">Marcell Jacobs</th>
33      <td>2 medaglie d'Oro</td>
34      <td colspan="2">100<abbr title="metri">m</abbr>
35        maschile, staffetta 4x100
36      <abbr title="metri">m</abbr></td>
37    </tr>
38    <tr>
39      <th scope="row">Vanessa Ferrari</th>
40      <td>1 medaglia d'argento</td>
41      <td colspan="2">Ginnastica Artistica, corpo libero</td>
42    </tr>
43    <tr>
44      <th scope="row">Gianmarco Tamberi</th>
45      <td>1 medaglia d'oro</td>

```

15 Schema o struttura organizzativa

Gli schemi esatti sono i migliori quando l'utente sa quello che sta cercando.

Gli schemi ambigui sono i migliori per la navigazione e l'apprendimento associativo, quando l'utente ha una vaga idea di quello che sta cercando.

Quando possibile è opportuno utilizzare entrambi i tipi di schema.

Le principali strutture organizzative sono:

- Sequenza

- Gerarchia
- A faccette (a matrice)
- Ipertesto

15.1 Sequenza

La sequenza rappresenta il modo più semplice di organizzare l'informazione.
È adatto per siti didattici, perché impone un ordine al materiale da consultare.

15.2 Gerarchia

Una gerarchia ben progettata è la base di una buona architettura informativa.
Le suddivisioni mutuamente esclusive e le relazioni padre-figlio sono semplici e familiari.

Se usate per un sito web rendono gli utenti in grado di sviluppare facilmente un modello mentale della struttura del sito e della loro localizzazione in questa struttura. In questa struttura è necessario trovare l'equilibrio tra ampiezza e profondità, maggiori dettagli nella domanda [11](#).

15.3 Faccette o Matrici

Ogni informazione può essere vista da diverse angolature, perché ha molti attributi attraverso cui può essere conosciuta e usata.

Ad esempio una maglia può essere catalogata per taglia, tipologia, materiale, colore, ...

15.4 Ipertesto

Struttura organizzativa innovativa.

Le unità informative possono essere collegate attraverso link gerarchicamente o non gerarchicamente, oppure seguendo entrambe le modalità.

È una struttura non lineare, molto flessibile ma è un ostacolo per l'utente per la formazione di un modello mentale del sito.

Non si adatta bene per la navigazione primaria.

16 Si può capire con solo test automatici se un sito è accessibile

Non è possibile in quanto la maggior parte delle cose non sono controllabili tramite test automatici ma solo semiautomatici.

Alcuni esempi di cose non controllabili sono il layout corretto e visibile del sito oppure la correttezza del testo alternativo dentro l'attributo alt.

Relativamente a quest'ultimo si può però controllare la presenza o la mancanza.

17 Distinzione fra presentazione, struttura e comportamento

Struttura = (o contenuto) è il testo.

Presentazione = come appare il sito.

Si sviluppa con il CSS, decisamente meglio se in un file separato ma si può fare anche nello stesso file.

Comportamento è JavaScript, sarebbe meglio dividerlo, più mantenibile.

18 Nascondere, anche solo parzialmente, il contenuto dei menù tramite menù a tendina è corretto?

L'utilizzo del menu hamburger su desktop è generalmente sconsigliato perché compromette sia l'usabilità che l'accessibilità.

Nascondendo le opzioni di navigazione primaria dietro un'icona, si riduce drasticamente la loro visibilità e scopribilità, costringendo l'utente a un click aggiuntivo solo per visualizzare le possibili destinazioni.

Questo design peggiora l'esperienza di navigazione da tastiera, richiede maggiore precisione di puntamento e contraddice le aspettative consolidate degli utenti desktop, che si attendono una barra di navigazione sempre visibile e immediatamente utilizzabile.

19 Differenza tra convenzioni interne ed esterne

Le convenzioni esterne sono gli standard universali del web che gli utenti conoscono (esempio: il logo in alto a sinistra rimanda alla home). Non vanno infrante se non per un beneficio davvero straordinario, per non disorientare l'utente.

Le convenzioni interne sono le regole di coerenza che stabiliamo all'interno del nostro sito (ad esempio, tutti i pulsanti di conferma sono verdi). Vanno sempre rispettate per aiutare l'utente a costruire una mappa mentale affidabile dell'interfaccia e navigare con sicurezza.

20 Mettere in ordine di applicazione i seguenti stili CSS

1. Impostazioni di stile predefinite del browser
2. Impostazioni personali dell'utente
3. Fogli di stile esterni definiti dall'autore
4. Fogli di stile embedded (<style>) definiti dall'autore
5. Impostazioni di stile inline (style="") definite dall'autore della pagina

21 Qual è il media più accessibile?

Il testo, in quanto può essere letto da screen reader, tradotto e ingrandito.

22 Quali sono le tra domande più importanti alle quali devo saper rispondere, pena il fenomeno del disorientamento?

- Dove sono? In titolo e intestazione
- Dove posso andare? In menù di navigazione
- Di cosa si tratta? Nel contenuto della pagina