

UNI COOKED



AUTORI

Andrea Precoma	2068227
Andrea Soranzo	2075539
Gabriele Magnelli	2075542
Matteo Schievano	2075537

caa.studenti.math.unipd.it/asoranzo

DATI DI ACCESSO

Nome utente	Password	Tipo
user	user	DAD
GGGaggi	GGGaggi	Pendolare
Tony	Tony	In sede
UgoFantozzi	UgoFantozzi	Fuorisede

Referente

Andrea Soranzo

andreajesus.soranzomendez@studenti.unipd.it

INDICE

1. Abstract	2
1.1. Metodologia di sviluppo	2
2. Accessibilità	2
2.1. Caratteristiche generali	2
2.2. Controlli ulteriori	3
2.3. Colori	3
2.4. Strumenti utilizzati	4
2.5. Falsi positivi e warning	4

1. Abstract

Il progetto consiste in un sito web che tratta un ricettario dedicato agli studenti universitari.

Vi sono quattro categorie di universitario per differenziare le esigenze di ciascuno: queste vengono utilizzate anche per suddividere le ricette considerando la stima del costo, la difficoltà e il tempo di preparazione.

La ricetta può essere valutata dagli utenti in una modalità simile da una prova d'esame: viene assegnata un voto da 1 a 30 e un commento che motiva la valutazione.

Ogni utente registrato possiede un proprio profilo con dei dati che può personalizzare. Le ricette salvate come preferite sono visibili nella pagina profilo.

Nella pagina dedicata ai contatti dei proprietari del sito è presente anche un campo di testo nel quale ogni utente può suggerire un piatto da aggiungere al sito.

1.1. Metodologia di sviluppo

Nonostante la maggior parte dell'utenza derivi dal mobile, abbiamo deciso di cominciare dallo sviluppo del sito desktop per facilitare la fase di design. Dovendo comunque completare la visualizzazione sia per desktop che per mobile, ci è sembrato più utile adottare un approccio che agevolasse il lavoro pur mantenendo il medesimo risultato.

2. Accessibilità

2.1. Caratteristiche generali

Per soddisfare i requisiti di accessibilità abbiamo adottato le seguenti misure:

- Aggiunto la breadcrumb in ogni pagina.
- Aggiunti gli aiuti alla navigazione in ogni pagina. I "torna su" sono posizionati sempre nel footer ma anche in sezioni lunghe, quindi tra le categorie della home e nella pagina delle ricette.
- Contrassegnato abbreviazioni, simboli e sigle con i corretti tag HTML5.
- Aggiunto l'attributo title ai link in modo da renderli più espressivi alla lettura con screen reader.
- Posizionato gli input e le relative label seguendo le indicazioni delle [WCAG 2.1](#), comunque sempre prima dell'eventuale messaggio di errore.
- Scelto i colori per sfondo, testo, link visitati e non visitati in almeno contrasto AA. Per distinguere i link visitati da quelli non visitati, non potendoci basare sui colori, abbiamo provveduto a cambiare lo stile di sottolineatura.
- Assegnato gli attributi alt a tutte le immagini lasciandoli vuoti se non necessari.
- Aggiunto l'attributo aria-label dove necessario.
- Nascosto gli elementi di aiuto agli screen reader.
- Eseguito l'image replacement dove necessario.
- Assegnato gli attributi aria-live e role per avvisare lo screen reader di un cambiamento.

Il menu ad hamburger presente nel sito mobile è reso accessibile mediante una label che cambia testo dinamicamente a seconda dello stato del menu. Quando il menu è aperto la porzione di sito visibile al di sotto viene offuscata in modo da impedire l'interazione con il sito e concentrare l'attenzione dell'utente sul menu. Viene inoltre bloccato lo scorrimento verticale sfruttando la pseudo-classe :has(): se questa non è supportata dal browser in uso, l'utente può scorrere nella pagina ma questa rimane offuscata, obbligandolo a chiudere il menu. L'utente che adopera lo screen reader può navigare senza troppi problemi nel sito anche quando il menu è aperto. Da notare infine che un utente con difficoltà visiva che utilizza lo screen reader con molta probabilità non è infastidito dal filtro che offusca le pagine.

Seguendo le indicazioni delle [WCAG 2.1](#) gli input di tipo checkbox precedono le label, in tutti gli altri casi avviene il contrario. Unica eccezione riguarda l'input per modificare la foto profilo: questo deve essere posto all'interno della label per permetterne il corretto funzionamento tramite regole CSS.

I valori degli slider nei filtri della pagina che visualizza tutte le ricette sono nascosti allo screen reader: nei test svolti questi vengono letti ogni volta che si aggiornano, dunque abbiamo ritenuto superfluo l'inserimento di un eventuale tag alert. Il minimo e massimo sono esplicitati nella label.

I link "vai alla ricetta" degli elementi della lista potrebbero risultare poco espressivi per un utente che usa lo screen reader: per ovviare al problema abbiamo aggiunto l'attributo `title` con il nome della ricetta. Situazione simile per alcune `label`, ad esempio "mostra password", alle quali è stato assegnato l'attributo `aria-label` così da assicurarsi che venga letto dallo screen reader indipendentemente dalle impostazioni personali.

Alla pagina di una ricetta si può arrivare dal catalogo delle ricette oppure dall'elenco delle ricette preferite di un utente: per migliorare l'esperienza nell'uso della breadcrumb, questa cambia pagina di provenienza a seconda che si arrivi dalla pagina delle ricette o da un preferito.

La sezione delle valutazioni nella pagina della singola ricetta può potenzialmente diventare molto lunga: abbiamo quindi provveduto a limitare il numero di valutazioni visualizzate aggiungendo un pulsante per caricarne altre. Al termine delle valutazioni è posto anche un link per tornare rapidamente all'inizio della sezione. In ogni valutazione inoltre è presente il nome dell'utente sotto forma di link: in questo modo è possibile visitare il profilo degli altri utenti.

Riguardo la potenziale numerosità delle ricette preferite abbiamo adottato la stessa soluzione utilizzata per i commenti nelle ricette.

Una nota di riguardo è dovuta al caricamento delle foto: nonostante le pagine HTML e le componenti JavaScript e PHP siano appositamente configurate per supportare solo il caricamento di file `.jpeg`, `.jpg` e `.png`, talvolta i dispositivi Apple non considerano questi limiti e permettono di caricare qualsiasi foto presente nel rullino del dispositivo che, solitamente, sono in formato `.heif` o `.heic`. PHP nativamente non supporta la conversione di tali formati in `.webp`, formato che garantisce maggiore efficienza nel caricamento delle pagine web. Inoltre, nessun browser, ad eccezione di Safari, permette la visualizzazione di tali immagini: per questi motivi, il sito web non permette il caricamento di file di tipo `.heif` o `.heic`.

2.2. Controlli ulteriori

Sono stati eseguiti i seguenti controlli per quanto riguarda l'accessibilità:

- Tutti gli elementi interattivi sono raggiungibili tramite "tab" o "swipe".
- Tutti gli input sono preceduti dalle label e seguiti dall'eventuale errore (ad eccezione del menu ad hamburger).
- Tutte le pagine contengono la breadcrumb.
- Attributo `alt` ad ogni immagine (ove l'informazione non sia fornita dal contesto circostante).
- Attributo `o tag abbr` ad ogni abbreviazione o sigla.
- Link corretti per gli aiuti alla navigazione.
- Attributo `lang` con la lingua dei termini che seguono se sono stranieri (i contenuti direttamente pubblicati degli utenti, tuttavia, non possono essere contrassegnati in quanto «dinamici» e non sottoposti a controllo da parte degli amministratori del sito web)
- Resistenza degli elementi con ingrandimento del testo fino a 24px.

2.3. Colori

Sono stati eseguiti i seguenti controlli per quanto riguarda i contrasti dei colori:

- Tutti i colori di sfondo e testo sono in contrasto AAA, sia nel tema chiaro che quello scuro. Uniche eccezioni sono:
 - per il **tema chiaro**: i link nel menu e i link nel testo, entrambi comunque in contrasto AA e distinguibili per via della sottolineatura.
 - per il **tema scuro**, il testo "Adatto per te se:" nel lato posteriore delle carte nella home. In base a quanto riportato da WCAG color contrast checker l'intestazione è in contrasto AA in quanto considerato come testo grande. Perde il contrasto AA quando il testo 1em raggiunge i 10px, tuttavia riteniamo che tali numeri non siano adottati da un utente con difficoltà nella distinzione dei colori.
- Mantenendo i colori in palette non è stato possibile trovare i contrasti tra link visitati e non visitati (mantenendo i contrasti tra questi e lo sfondo). Abbiamo quindi optato per due differenti soluzioni:
 - I link inseriti nel testo o in elenchi sono sottolineati una volta se non visitati, due volte se visitati. Si fa notare che il colore dei link non visitati varia a seconda della locazione degli stessi: non assumono cioè colore bianco se si trovano in prossimità di testo normale in modo da risaltare maggiormente.

- I link con uno sfondo dedicato, ad esempio i "pulsanti" ACCEDI nel menu o MODIFICA PROFILO nel profilo utente, sono contornati da un bordo dello stesso colore del link se visitati. Si fa notare che il colore dei link rimane lo stesso perché il bordo aggiunto è sufficiente per distinguere i due stati.
- È stato assegnato un colore di sufficiente contrasto allo sfondo di ciascun testo. In questo modo si assicura una corretta lettura anche nel caso remoto in cui il browser non riesca a caricare le immagini.

2.4. Strumenti utilizzati

Per eseguire i controlli sono stati adottati i seguenti strumenti:

- **Total Validator v18.2.0** per assicurare la correttezza dei documenti .html (completati da JavaScript e PHP).
- **W3C CSS Validator**
- **WCAG color contrast checker** (estensione Chrome)
- Screen reader:
 - **NVDA**
 - **Narrator** (Windows 11)
 - **Orca** (Ubuntu 24.10)
- **Lighthouse**

Molti controlli sono stati svolti manualmente.

2.5. Falsi positivi e warning

Validando il sito con la versione 18.2.0 di Total Validator vengono evidenziati i seguenti falsi positivi e warning seguiti dalle motivazioni:

- Le label sono state posizionate prima degli input per questioni di accessibilità, eccezione fatta per l'input di tipo checkbox e radio (si veda [WCAG 2.1](#)). L'input usato per modificare la foto profilo dovrebbe essere posizionato dopo la label secondo le WCAG, così però richiederebbe l'uso della pseudo-classe :has(). Per renderlo funzionante con qualsiasi versione del browser abbiamo preferito mettere l'input dentro la label, soluzione comunque valida e discussa con la professoressa Gaggi.
- Nella home, il primo link di aiuto alla navigazione non è quello per andare al contenuto, bensì quello che porta al menù contenente i link per andare velocemente alla descrizione delle varie tipologie di utente: abbiamo preferito fosse questo il primo link di aiuto per poter accedere velocemente alle varie descrizioni, ma Total Validator segnala la cosa come avvertimento (ma non errore).
- Nella home la label "scopri se fa per te" è ripetuta perché l'azione è la medesima per tutte le quattro sezioni.
- Nella pagina per effettuare la registrazione, la label per la checkbox «Mostra password» viene segnalata come ripetuta: non è stata inserita una legend differente perché il campo su cui è rivolta l'azione è necessariamente quello che è stato appena popolato dall'utente seguendo l'ordine della pagina. Stesso errore e stessa motivazione è data per i «Mostra password» nella sezione di cambio password presente nella pagina per la modifica del profilo.
- Nella pagina di una ricetta, se l'accesso è stato eseguito e l'utente può lasciare una valutazione, il form ha un fieldset senza legend: Total Validator segnala con un avvertimento, tuttavia l'informazione che la legend dovrebbe fornire è in realtà fornita dal titolo dell'intera sezione (**Valutazioni**) e dalla prima label (**Valuta questa ricetta**). L'inserimento dunque di una legend potrebbe rendere dunque meno piacevole la consultazione del sito web.
- Nella pagina delle ricette la form per l'ordinamento dei risultati non contiene il submit perché l'aggiornamento della vista è automatico alla modifica del valore della select.
- Per la ragione citata precedentemente, PHP deve inserire onchange="this.form.submit()".
- Nella pagina dei contatti non è necessaria una legend per la form dei suggerimenti perché è composta solo da una textarea che possiede la propria label.
- Nella pagina di modifica profilo non è necessaria una legend per la form di modifica password perché il compit è esplicito nel titolo e nelle label.

Validando i fogli di stile con W3C CSS Validator vengono evidenziati i seguenti falsi positivi seguiti dalle motivazioni:

- le regole "webkit" e "moz" sono state aggiunte per migliorare il più possibile l'esperienza dell'utente su qualsiasi browser.
- le clausole "selector" sono risultate necessarie per garantire il funzionamento del sito su qualsiasi browser.