

Progetto di Tecnologie Web

Fuji Sushi

Informazioni

Componente 1	Diego Stafa 1226285
Componente 2	Luciano Wu 1224452
Componente 3	Samuele Rizzato 1226307
Anno	2021/22
Destinatari	Prof.ssa. Ombretta Gaggi

Sommario

Relazione del progetto di Tecnologie Web.

Indice

1	Introduzione	4
1.1	Abstract	4
1.2	Analisi dell'utenza	4
2	Progettazione	5
2.1	Struttura del sito	5
2.1.1	Header	6
2.1.2	Contenuto	6
2.1.3	Footer	7
2.1.4	Base di dati	7
2.1.4.1	Schema concettuale	7
2.1.4.2	Schema logico	8
2.1.5	Ricerche probabili	8
3	Realizzazione	9
3.1	Linguaggi	9
3.2	Strumenti	9
4	Accessibilità	9
4.1	Separazione tra comportamento, struttura e presentazione	10
4.2	WAI-ARIA	10
4.3	Colori	10
4.3.1	Test dei colori	11
4.4	Noscript	19
4.5	Tabindex	19
5	Presentazione	19
5.1	Desktop	19
5.2	Tablet	20
5.3	Mobile	21
5.4	Stampa	22
6	Comportamento	23
6.1	Javascript	23
6.1.1	Funzionalità	23
6.2	PHP	24
6.2.1	Validazione	24
6.2.2	Sicurezza	24
6.2.3	Sessioni	24
6.2.4	Classe SushiDB	25
7	Testing	25
7.1	Validazione	25
7.2	Peso	25
7.3	Browser e dispositivi	25

8	Organizzazione del gruppo	26
8.1	Processo di sviluppo	26
8.2	Divisione del lavoro	26
8.2.1	Diego Stafa	26
8.2.2	Luciano Wu	26
8.2.3	Samuele Rizzato	26

Informazioni sul sito

Indirizzo del sito web <http://tecweb.studenti.math.unipd.it/srizzato/>

Referente del gruppo: Samuele Rizzato 1226307

Mail del referente: samuele.rizzato.1@studenti.unipd.it

Email MyWcag4All: luciano.wu@studenti.unipd.it

Account MyWcag4All: luciano

Tipologia di utente	Username	Password
Amministratore	admin	admin
Utente generico	user	user

Tabella 1: Credenziali degli utenti del sito.

1 Introduzione

1.1 Abstract

Fuji Sushi è un piccolo ristorante dai sapori orientali che sta rapidamente conquistando il mercato, al fine di consolidare e mantenere questa scia di successo il ristorante ha deciso di rinnovare il proprio sito *web* per l'acquisto e recapito dei loro deliziosi piatti.

Il nuovo sito vuole essere un passo avanti alla concorrenza, la quale solitamente espone solo un catalogo statico del loro menù, perciò *Fuji Sushi* mette a disposizione un vasto repertorio di funzionalità per tutti i suoi utenti. La funzionalità cardine riguarda la visualizzazione e l'acquisto dei piatti offerti dal ristorante, ma il sito offre anche molte altre comodità, tra cui:

- un sistema di filtraggio per il vasto menù;
- un *feed* delle ultime notizie del ristorante. per rimanere sempre aggiornati sulle ultime novità;
- un servizio di *take-away*, con la possibilità di ordinare ad orari flessibili e programmati a discrezione dell'utente;
- un carrello intelligente che ricorda le scelte del cliente, anche se non autenticato;
- un sistema di notifica per dare un *feedback* all'utente delle sue azioni;
- molte altre azioni per gli utenti autenticati.

Oltre ad essere funzionale il sito vuole anche essere informativo, fornendo una pagina per i contatti rilevanti e una pagina che suggerisce i migliori piatti del momento.

Nonostante il sito ruoti attorno all'utente, l'autenticazione non è obbligatoria: i visitatori hanno comunque il pieno accesso al sito, al menù e anche al carrello, ma ovviamente non sono abilitati all'acquisto. Infine il sito segue una serie di convenzioni e idee progettuali che si sono rivelate di successo nel tempo, quali:

- rispetto degli standard *W3C*, *HTML5* e *CSS3*;
- netta separazione tra struttura, presentazione, comportamento;
- adesione alle regole di accessibilità, con gli standard *WAI-ARIA* e *WCAG* (AAA).

1.2 Analisi dell'utenza

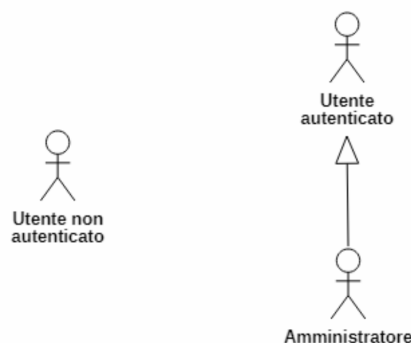


Figura 1: Gerarchia degli utenti.

Il sito web si rivolge prevalentemente agli amanti del sushi, le funzionalità sono state pensate per favorire i diversi bisogni dell'utente durante l'ordine, ponendo l'attenzione sulla flessibilità degli acquisti, tramite un servizio di recapito o *take-away* e una pianificazione del momento di ritiro.

L'utente generico non autenticato deve poter anche trovare le ultime notizie riguardanti il ristorante, per portarlo a conoscenza di nuovi piatti o eventi, e visualizzare i piatti del momento tramite un *ranking* dei piatti più richiesti.

Il sito è stato anche pensato per l'uso da parte dei gestori, attraverso un sistema di gestione del menù per l'aggiunta o rimozione di piatti e categorie.

L'utenza è così suddivisa:

- **Utente non autenticato:** ricerca informazioni, visita il menù e le *news*, può aggiungere piatti al carrello ma non può finalizzare gli ordini;
- **Utente autenticato:** ha accesso a tutte le funzionalità di un utente non autenticato, inoltre è abilitato all'acquisto dei piatti scelti e alla gestione del suo profilo personale;
- **Amministratore:** ha accesso a tutte le funzionalità di un utente autenticato, inoltre gestisce il menù e le *news* del ristorante.

Per quanto riguarda il lato tecnologico, il sito è anche pensato per essere fruibile da dispositivi variegati, implementando quattro diverse viste del sito:

- visualizzazione *desktop*;
- visualizzazione *tablet*;
- visualizzazione *mobile*;
- visualizzazione *desktop*.

2 Progettazione

2.1 Struttura del sito

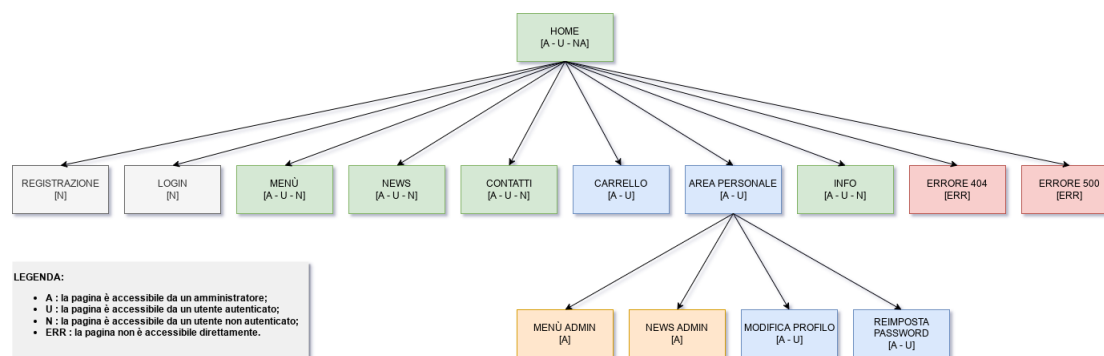


Figura 2: Mappa che illustra la struttura del sito.

Il sito è basato su una struttura sia gerarchica che a ipertesto, ogni pagina può accedere ad un'altra pagina sullo stesso livello, tramite degli ipertesti, ma per scendere di livello è necessario passare per la pagina di riferimento. I collegamenti forniti dipendono dai permessi dell'utente, in ciò un utente non autenticato vede quindi una struttura del sito semplificata e di più facile navigazione.

Il sito è stato strutturato seguendo un approccio vicino all'utente, seguendo lo schema a tre pannelli:

- **Dove sono?:** l'informazione è reperibile nel breadcrumb;
- **Dove posso andare?:** l'informazione è reperibile nella barra superiore di navigazione.
- **Di cosa si tratta?:** l'informazione è reperibile nel titolo della pagina e in quelli delle sezioni;

2.1.1 Header



Figura 3: Header

L'*header* è l'elemento di riferimento per la navigazione del sito, è composto di più parti:

- il menù di navigazione a sinistra per spostarsi nelle pagine accessibili da ogni tipologia di utente, visualizzabile in modalità *desktop* e accessibile dal menù a comparsa in modalità *mobile*;
- il menù di navigazione a destra per spostarsi nelle pagine accessibili dagli utenti autenticati o per eseguire l'accesso, visualizzabile in modalità *desktop* e accessibile dal menù a comparsa in modalità *mobile*;
- il titolo del sito, composto sia dal logo che dal nome;
- il *breadcrumb*, che indica il percorso effettuato per accedere alla pagina in cui ci si trova, fornendo anche i *link* per tornare alle pagine precedentemente visitate;
- il *link* per spostare il focus sul contenuto della pagina.

2.1.2 Contenuto

Di seguito le pagine che compongono il sito:

- **Home:** è la pagina principale del sito, offre uno sguardo veloce sulle ultime *news* del ristorante e sui piatti del momento, in fondo alla pagina è presente inoltre una breve sezione sulla storia del ristorante e delle funzionalità del sito;
- **Menù:** contiene il catalogo completo dei piatti, divisi per categoria, che l'utente può aggiungere al carrello temporaneo. Il carrello nel menù permette di: salvare il carrello permanentemente, svuotare il carrello, rimuovere dei piatti o modificarne la quantità ordinata;
- **News:** contiene una lista di tutte le notizie del ristorante, dalla più recente alla più vecchia. Ogni *news* è composta di un titolo, un contenuto, una data di pubblicazione e un'immagine rappresentativa;
- **Contatti:** contiene i recapiti, gli orari di lavoro e ulteriori contatti per la comunicazione diretta con il ristorante;
- **Carrello:** visualizza il carrello corrente dell'utente e permette di completare l'ordine;
- **Area personale:** punto di ingresso per la gestione del proprio profilo (per l'utente autenticato) o per la gestione del sito (esclusivo per l'amministratore). L'area personale contiene anche lo storico degli ordini effettuati, visualizzato in formato tabellare;
- **Login:** contiene il modulo di autenticazione;

- **Registrazione:** contiene il modulo di registrazione;
- **Menù *admin*:** permette di aggiungere ed eliminare piatti o categorie dal menù;
- **News *admin*:** permette di aggiungere ed eliminare;
- **Modifica profilo:** permette di modificare i dati personali dell'utente;
- **Reimposta *password*:** permette di modificare la password dell'utente;
- **Errore 404:** pagina informativa, indica che la risorsa richiesta non esiste;
- **Errore 500:** pagina informativa, indica che il *server* non è in grado di soddisfare la richiesta;
- **Info:** pagina informativa, indica l'esito di una determinata azione dell'utente.

2.1.3 Footer

All'interno del footer compaiono elementi di generica utilità e informazioni importanti che non sono presenti nel resto del sito:

- i contatti del ristorante;
- le tecnologie e gli standard utilizzati per comporre il sito;
- gli autori e proprietari del sito.

2.1.4 Base di dati

2.1.4.1 Schema concettuale

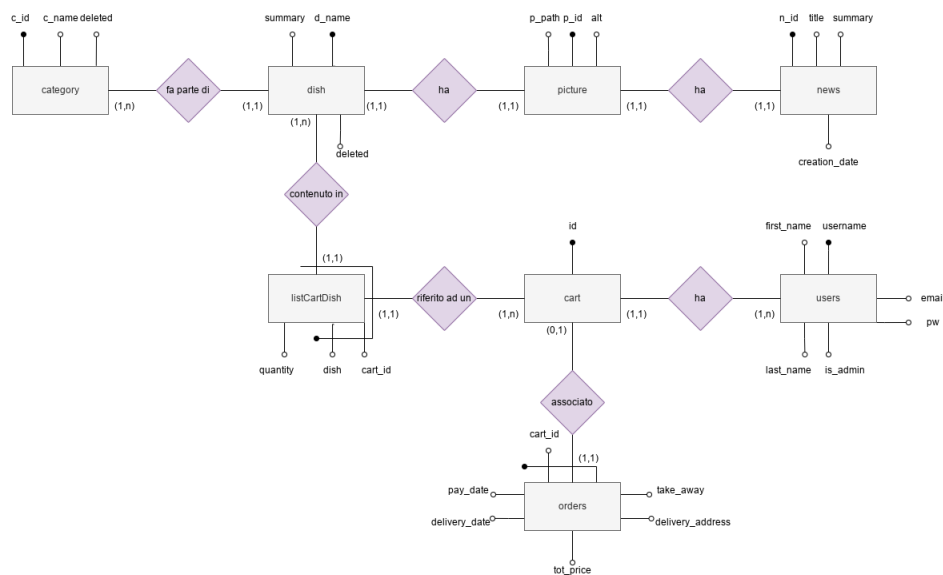


Figura 4: Schema concettuale del database.

La base di dati è stata progettata per supportare le funzionalità offerte dal sito, consentendo di:

- salvare le informazioni di un utente;

- contenere i piatti del menù;
- inserire ed eliminare nuove categorie;
- inserire ed eliminare news;
- creare ordini;
- mantenere lo stato del carrello.

La base di dati inoltre è stata scomposta in terza forma normale di Boyce Codd per renderla manutenibile nel tempo.

2.1.4.2 Schema logico

```
category (c_id, c_name, deleted)
picture (d_name, p_path, alt)
dish (p_id, summary, price, deleted, category, picture)
users (username, email, first_name, last_name, pw, is_admin)
news (n_id, title, summary, creation_date, picture)
cart (id, users)
orders (cart_id, pay_date, delivery_date, tot_price, take_away, delivery_address)
listCartDish (cart_id, dish, quantity)

dish.category -> category.c_id
dish.picture -> picture.p_id
news.picture -> picture.c_id
cart.users -> users.username
orders.cart_id -> cart.id
listCartDish.dish -> dish.d_name
listCartDish.cart_id -> cart.id
```

2.1.5 Ricerche probabili

Le pagine contengono la seguente lista di parole chiave per facilitare e indirizzare la ricerca dell'utente al nostro sito:

- Fuji;
- sushi;
- ristorante;
- giapponese;
- takeaway;
- menu;
- piatti;
- cucina;
- pesce.

3 Realizzazione

3.1 Linguaggi

- **HTML5:** per la struttura del sito, HTML5 offre tag utili per una scrittura semantica delle pagine (header, main, footer...) e molti attributi per controlli *built-in*, come il *pattern* o *required*, e di accessibilità, come gli attributi *ARIA*;
- **CSS3:** utilizzato nella parte di presentazione, per costruire un layout *responsive* e fluido, la terza versione offre i nuovi *grid* e *flex* per organizzare il *layout* del menù e struttura le sezioni delle pagine;
- **JavaScript:** per la validazione dei *form* lato client, per mantenere lo stato locale del carrello dell'utente, per il menù e carrello a compare. Non è stata utilizzata alcuna libreria esterna e tutte le funzionalità sono state inserite in un unico file per ridurre il tempo di latenza dell'utente;
- **PHP:** utilizzato per gestire la logica di *business* del sito e realizzare pagine *web* dinamiche, per le interazioni col database e la validazione lato *server*;
- **SQL:** utilizzato per interagire con la base di dati, tramite *query* per l'inserimento/rimozione/aggiornamento dei dati.

3.2 Strumenti

- **Google Meet:** per gestire gli incontri del gruppo;
- **GitHub:** per collaborare in modo efficace tra più persone;
- **Server PHP:** per eseguire il sito in locale e testare le sue funzionalità;
- **MySqlWorkbench:** per avere un'interfaccia sullo stato della base di dati;
- **MyWcag4All:** per eseguire controlli multipli di accessibilità;
- **Wave Extension:** per controllare errori strutturali del sito (link mancanti, immagini senza alt...).
- **WCAG Color Contrast Checker:** per essere conformi allo standard WCAG AAA per il contrasto dei colori usati;
- **HTML outliner:** per evidenziare errori nella struttura del codice HTML;
- **HTML5 Validator:** per conformare il codice allo standard HTML5;
- **CSS3 Validator:** per conformare il codice allo standard CSS3.

4 Accessibilità

Lo standard su cui ci siamo conformati è WCAG 2.0 (<https://www.w3.org/Translations/WCAG20-it/>)

4.1 Separazione tra comportamento, struttura e presentazione

La separazione tra i diversi aspetti del sito è stata perseguita dividendo in tre parti:

- parte strutturale: con i file *HTML*, definiscono la struttura e le componenti delle pagine in modo statico;
- parte di presentazione: con i file *css*, definiscono l'aspetto delle pagine e delle loro componenti;
- parte comportamentale: con i file *javascript*, definiscono le funzioni lato *client* che il sito offre (validazione, carrello locale e menù a comparsa).

All'interno dei file *HTML* non sono stati integrati tag inerenti alle parti di presentazione o comportamento, gli stili vengono applicati includendo i riferimenti a i file esterni.

4.2 WAI-ARIA

Nel progetto abbiamo deciso di utilizzare lo standard *WAI-ARIA*, più precisamente l'attributo *aria-label* è stato usato al fine di rendere accessibile il *breadcrumb* presente nell'*header* delle pagine. Per altri elementi o attributi abbiamo deciso di adottare quelli standard forniti da *HTML*.

4.3 Colori

Tutti i colori utilizzati superano il test textitWCAG AAA, ma non per alcuni tipi di cecità cromatica, dove si raggiunge solo lo standard AA, di seguito la *palette* usata;

- **dark-accent:** *#1f0707*, è una tinta di rosso scuro tendente al marrone ispirati dalla salsa di soia, usato per gli sfondi delle sezioni e per i bottoni, o per tutto ciò che ha come sfondo un *light-accent*;
- **light-accent:** *#ffffff*, è un bianco candido ispirato al riso, usato per lo sfondo del contenuto delle sezioni e per i bottoni inattivi, per tutto ciò che ha come sfondo un *dark-accent*;
- **bg-front:** *#ffa991ef*, è un rosa salmone usato per lo sfondo principale delle pagine (*tag main*);
- **bg-item:** *#ffeee4*, ispirati dagli gnocchi di riso, usato per lo sfondo degli elementi visualizzati (piatti e *news*);
- **text-light:** *#ffffff*, per i testi con sfondo scuro;
- **text-dark:** *#2f323a*, per i testi con sfondo chiaro.

4.3.1 Test dei colori

WCAG - Contrast checker

Level: **AAA** color-blindness: **none** Auto-refresh: **on**

Contrast	Size	Elements
✓ 9.14	small	2 a
✓ 9.2	small	3 p
✓ 10.04	large	1 button
✓ 11.36	small	20 ▶[li, h2, p, span, div]
✓ 12.82	small	5 p
✓ 17.04	small	3 a
✓ 19.24	small	19 ▶[a, button, li]
✓ 19.24	large	11 ▶[a, h1, span]
✓ 21	small	2 span

57 visible 9 hidden

check contrast

Foreground color (hex.): **#000000** Background color (hex.): **#FFFFFF**

Size	Contrast	AA	AAA
small	21	✓	✓
Large	21	✓	✓

Figura 5: Contrasto dei colori in assenza di cecità cromatica.

WCAG - Contrast checker

Level: **AAA** | color-blindness: **achromatomaly** | Auto-refresh: **on**

Contrast	Size	Elements
✓ 9.21	small	3 p
✓ 10.01	large	1 button
✓ 11.24	small	2 a
✓ 11.4	small	20 ▶[li, h2, p, span, div]
✓ 12.82	small	5 p
✓ 17.19	small	3 a
✓ 19.33	small	19 ▶[a, button, li]
✓ 19.33	large	11 ▶[a, h1, span]
✓ 21	small	2 span

57 visible 9 hidden

check contrast

Foreground color (hex.): **#000000** | Background color (hex.): **#FFFFFF**

Size	Contrast	AA	AAA
14pt	21	✓	✓
14pt bold	21	✓	✓
18pt	21	✓	✓

Figura 6: Contrasto dei colori per gli affetti da acromatomalia.

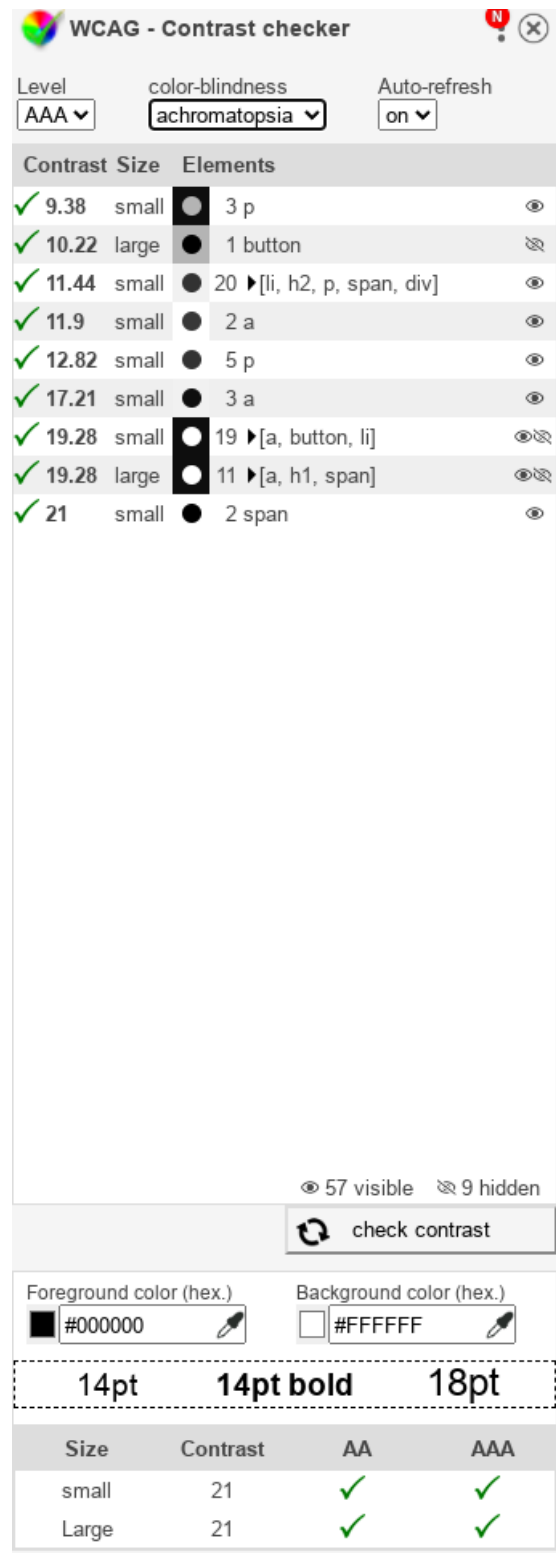


Figura 7: Contrasto dei colori affetti da acromatopsia.

WCAG - Contrast checker

Level: **AAA** color-blindness: **deuteranomaly** Auto-refresh: **on**

Contrast	Size	Elements	
✓ 10.02	small	3 p	👁
✓ 10.98	small	2 a	👁
✓ 11.09	large	1 button	👁
✓ 11.69	small	20 ▶[li, h2, p, span, div]	👁
✓ 12.92	small	5 p	👁
✓ 17.18	small	3 a	👁
✓ 18.99	small	19 ▶[a, button, li]	👁
✓ 18.99	large	11 ▶[a, h1, span]	👁
✓ 21	small	2 span	👁

👁 57 visible 🚫 9 hidden

check contrast

Foreground color (hex.): **#000000** Background color (hex.): **#FFFFFF**

	14pt	14pt bold	18pt
Size			
Contrast			
AA			
AAA			
small	21	✓	✓
Large	21	✓	✓

Figura 8: Contrasto dei colori per gli affetti da deuteronomalia.

WCAG - Contrast checker

Level: **AAA** color-blindness: **protanomaly** Auto-refresh: **on**

Contrast	Size	Elements
✓ 10.42	small	3 p
✓ 11.25	small	2 a
✓ 11.64	large	1 button
✓ 11.82	small	20 ▶[li, h2, p, span, div]
✓ 12.95	small	5 p
✓ 17.16	small	3 a
✓ 18.8	small	19 ▶[a, button, li]
✓ 18.8	large	11 ▶[a, h1, span]
✓ 21	small	2 span

57 visible 9 hidden

check contrast

Foreground color (hex.): **#000000** Background color (hex.): **#FFFFFF**

Size	Contrast	AA	AAA
14pt	21	✓	✓
14pt bold	21	✓	✓
18pt	21	✓	✓

Figura 9: Contrasto dei colori per gli affetti da protanomialia.

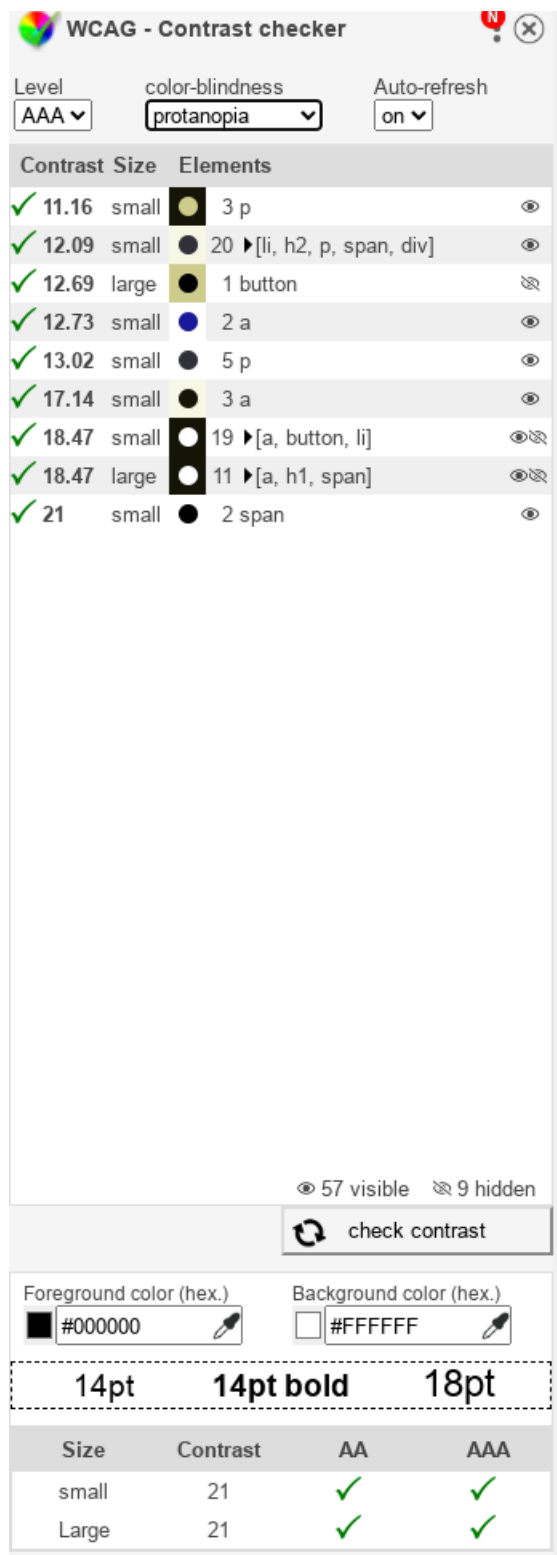


Figura 10: Contrasto dei colori per gli affetti da protanopia.

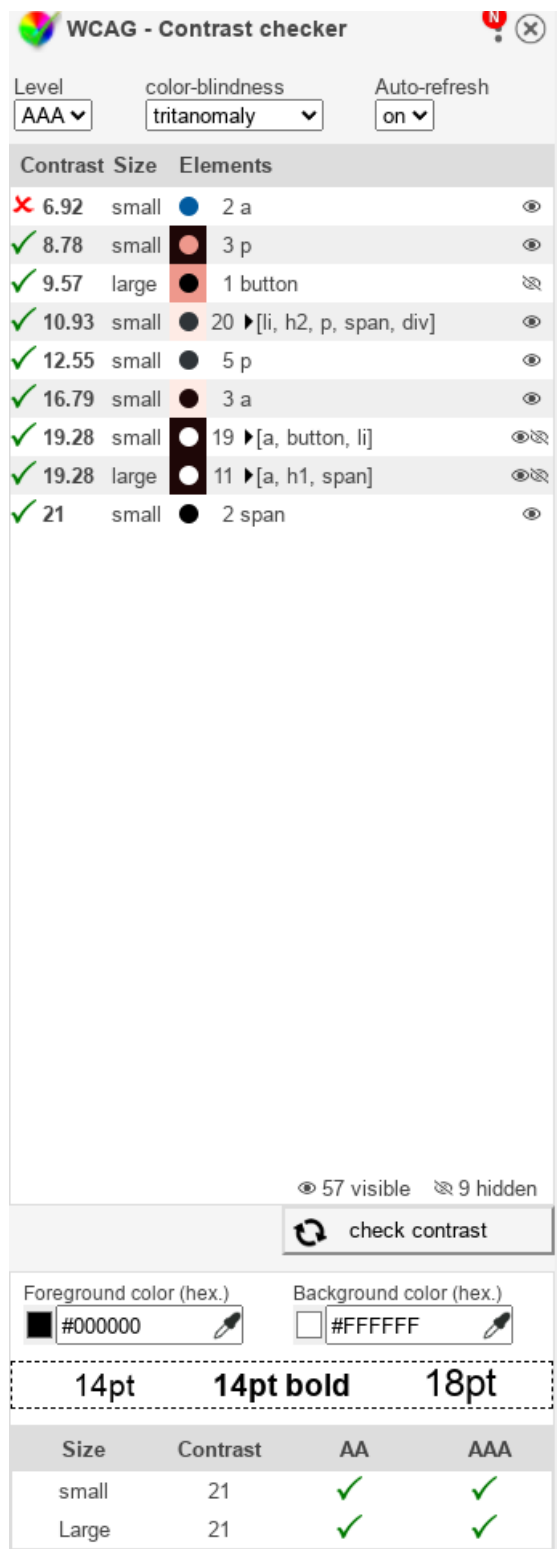


Figura 11: Contrasto dei colori per gli affetti da tritanomalia.



Figura 12: Contrasto dei colori per gli affetti da tritanopia.

4.4 Noscript

L'elemento *Noscript* è stato utilizzato per adattare elegantemente il sito a dispositivi che non hanno *JavaScript* abilitato, in particolare è stato usato per:

- visualizzare una forma alternativa del menù a comparsa in formato *mobile*.
- informare l'utente che la funzionalità del carrello locale è disattivata.

4.5 Tabindex

L'utilizzo dell'attributo *tabindex* è stato ridotto al minimo in quanto per la navigazione è naturalmente implementata dalla corretta struttura delle pagine.

l'unico punto in cui è stato utilizzato il *tabindex* è stato per saltare al contenuto senza dover percorrere tutti *link* presenti nell'*header*.

5 Presentazione

Per il sito è stato adottato un design fluido, con dei punti di rottura per i diversi dispositivi. Per quanto riguarda la presentazione, tramite *CSS* abbiamo implementato un design fluido e scalabile evitando il più possibile misure fisse, favorendo l'uso di misure relative alle preferenze dell'utente. Nella visualizzazione del menù è stato utilizzato layout *flex* e *grid* che garantiscono una trasformazione elegante nella presentazione dei piatti, che seguisse un layout a griglia.

5.1 Desktop

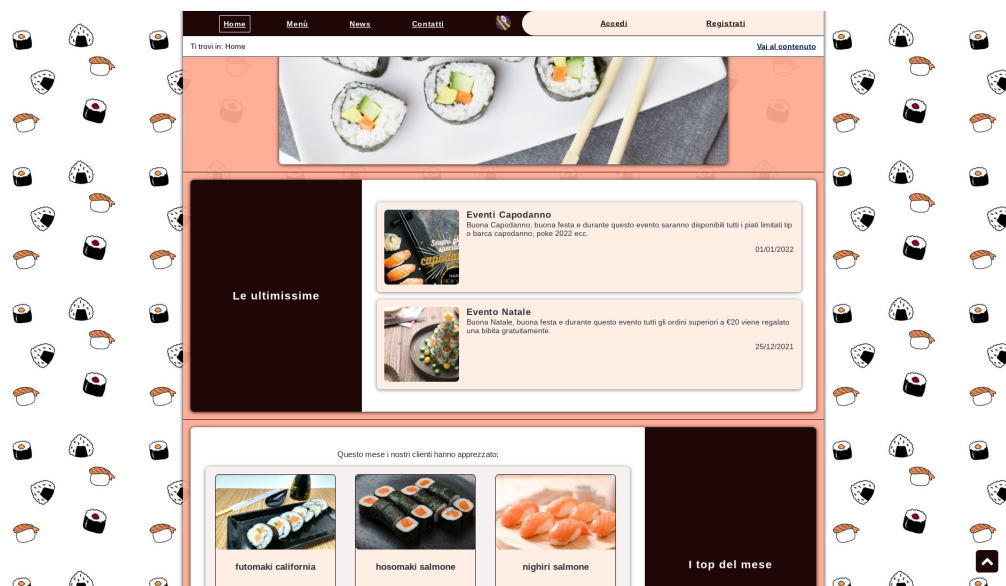


Figura 13: Layout desktop

5.2 Tablet



Figura 14: Layout tablet

5.3 Mobile



Figura 15: Layout mobile

5.4 Stampa

(tecweb/php/home.php)

Ti trovi in: [Home \(tecweb/php/home.php\)](#) > news

Tutte le notizie

Eventi Capodanno

Buona Capodanno, buona festa e durante questo evento saranno disponibili tutti i piatti limitati tipo barca capodanno, poke 2022 ecc.
01/01/2022

Evento Natale

Buona Natale, buona festa e durante questo evento tutti gli ordini superiori a €20 viene regalato una bibita gratuitamente.
25/12/2021

Evento Halloween

Durante il giorno di halloween sarà possibile acquistare piatti di limitati Halloween come il nighiri monster ecc.
31/10/2021

Evento Ferragosto

Durante questo evento tutti gli ordini superiori a €100 viene regalato un menù all you can eat gratuitamente.
15/08/2021

Evento Pasqua

Buona pasqua, buona cena e durante questo evento tutti gli ordini superiori a €10 viene regalato un uovo di Pasqua gratuitamente.
17/04/2021

Evento carnevale

Durante questo carnevale sarà possibile acquistare piatti di ogni genere, sia di pesce che di vegetariana a prezzo ridotto.
01/03/2021

Figura 16: Layout di stampa

Il *layout* di stampa mira a rimuovere gli elementi superflui (come footer, header, immagini di *layout* e bottoni rilevanti) e mantenere solo il contenuto della pagina, lasciando comunque informazioni essenziali per comprendere il contesto della pagina, come il nome, breadcrumb e titolo della pagina e sezione. Il *font* utilizzato nella stampa è il *serif*, per questioni di leggibilità, le immagini inoltre vengono ridotte al minimo indispensabile, mantenendolo solo per i piatti. Oltre a ciò il *link* visualizzano il percorso di arrivo, il sito poi adatta il layout in modo da renderlo piacevole anche in *layout* di stampa.

6 Comportamento

6.1 Javascript

Per il comportamento di determinate pagine del sito, lato *client*, è stato utilizzato un unico file *Javascript* che comprende diverse funzioni, la maggior parte per far segnalare gli errori nell'*input* agli *utenti*, in quanto il controllo di integrità dei dati è demandato alle funzionalità di *HTML5*, per quanto riguarda i controlli lato *client*, assegnando i giusti attributi ai tag *input*. Proprio per questo motivo nei vari *input* abbiamo utilizzato gli attributi *required*, *pattern*, *maxlength*, *min* e *max*. Inoltre, usando l'attributo *title*, quando è presente un errore nell'*input* e viene inviato il form, la pagina *HTML* di *default* fa visualizzare all'utente un messaggio di errore contenente la stringa dell'attributo *title*. Questo rende il codice Javascript relativo alla visualizzazione dei messaggi di errore una funzionalità aggiuntiva ma non essenziale. Siccome il solo controllo di integrità dei dati lato *client* non è sufficiente, tutte le nostre pagine *PHP* che ricevono dati in input ne controllano la validità lato *server* e in caso di non conformità fanno visualizzare un errore. In generale abbiamo fatto in modo che le funzioni *Javascript* non fossero essenziali per il completo funzionamento del sito e infatti le nostre pagine web sono completamente fruibili con tutte le loro funzionalità anche con Javascript disabilitato.

6.1.1 Funzionalità

- **initializeValidationForm()**: inizializza i controlli se nella pagina è presente un determinato *form*;
- **loading**: funzioni che assegnano i controlli ai vari elementi dei *form* delle pagine attraverso le funzioni **setOnBlurValidationRegex** e **setOnBlurConfirmPassword**;
- **Validation**: funzioni usate per controllare se certi elementi dei *form*, che non si possono validare solo con gli attributi HTML, siano stati inseriti correttamente altrimenti viene bloccato l'invio dei dati al server;
- **readOnlyAddress**: funzione utilizzata per impostare in sola lettura la via a cui consegnare l'ordine se viene scelta la modalità *take away*;
- **showErrorMessage**: mostra un messaggio d'errore sotto ad un campo del *form*;
- **resetErrorMessage**: rimuove un messaggio d'errore sotto ad un campo del *form*;
- **validateByRegex**: valida un campo del *form* attraverso una espressione regolare e nel caso non venga rispettata viene visualizzato un messaggio d'errore tramite **showError**;
- **validateDate**: valida la data e l'ora scelta dall'utente per la consegna dell'ordine e nel caso non venga rispettata viene visualizzato un messaggio d'errore tramite **showError**;
- **checkConfirmPassword**: controlla se la password di conferma è uguale alla password inserita in precedenza, in caso negativo visualizza un messaggio d'errore tramite **showError**;
- **confirmOrder**: conclude l'ordine corrente verificando che i dati dell'ordine siano validi e, se lo sono, prima di inviarli al server rimuove tutti i piatti da *local_storage*;
- **addCart**: aggiunge il piatto selezionato al carrello, aggiungendolo nel *local_storage*;
- **removeCart**: rimuove il piatto dell'indice dal carrello, rimuovendolo dal *local_storage*;
- **showCart**: mostra il carrello e il suo contenuto, creando una lista di piatti, le relative quantità e prezzi controllando il *local_storage*, nel caso non ci siano piatti nel carrello viene visualizzato un messaggio di carrello vuoto;
- **start**: funzione che inizializza il carrello, controllando il *local_storage* e i piatti nel carrello creati dal server, aggiorna il *local_storage* nel caso la lista di piatti sono diversi;

6.2 PHP

I file *PHP* gestiscono la logica di *business* del sito e sono strutturati nel seguente modo:

- un file *PHP* per ogni pagina del sito, che ne gestisce la logica;
- un file *PHP* per interfacciarsi alla base di dati sottostante;
- un file *PHP* per la validazione dei dati;
- un file *PHP* globale contenente le funzioni o variabili necessarie a tutte le altre pagine *PHP*;

6.2.1 Validazione

La validazione dei dati avviene in ambo le parti con delle espressioni regolari, nel *client* sia tramite gli attributi *built-in HTML5* di validazione, sia in *Javascript* il quale effettua anche una pulizia.

Nel *server* avviene un secondo controllo e pulizia dei dati inviati, ma anche un controllo logico, come ad esempio controllare che un utente esista già prima di crearne uno uguale.

6.2.2 Sicurezza

- ogni richiesta contenente dati sensibili, come quelle di autenticazione o acquisto, viene inoltrata con una *HTTP POST*;
- la pagina di registrazione non indica quale credenziale sia scorretta al fallimento dell'autenticazione;
- tutti i parametri inviati sono validati, controllati e ripuliti sia lato *client* che lato *server*;
- ogni *query* è eseguita con dei *prepared statement* per evitare *SQL injection*.

6.2.3 Sessioni

Le variabili di sessione php utilizzate sono le seguenti:

- `$_SESSION["logged"] = true` se l'utente è autenticato, `false` altrimenti;
- `$_SESSION["admin"] = true` se l'utente è autenticato è un amministratore, `false` altrimenti;
- `$_SESSION["info_msg"]` = contiene un messaggio di notifica per l'utente;
- `$_SESSION["redirect_link"]` = contiene il *link* in cui reindirizzare l'utente dopo aver visualizzato un messaggio di notifica;
- `$_SESSION["active_cart"]` = contiene l'identificativo del carrello corrente dell'utente autenticato;
- `$_SESSION["username"]` = contiene l'*username* dell'utente autenticato;
- `$_SESSION["first_name"]` = contiene il nome dell'utente autenticato;
- `$_SESSION["email"]` = contiene il cognome dell'utente autenticato;
- `$_SESSION["first_name"]` = contiene l'*email* dell'utente autenticato;

6.2.4 Classe SushiDB

Questa classe si occupa della comunicazione tra la base di dati sottostante e lo strato superiore di presentazione delle pagine, essa va a semplificare l'interazione e separa in modo netto le due parti, mettendo a disposizione le sole *query* necessarie per il sito.

SushiDB implementa inoltre diverse funzionalità utili:

- prepared statements per evitare *SQL injection*;
- lettura dei parametri di accesso alla base di dati da un file di configurazione esterno, ciò rende SushiDB indipendente dalle credenziali usate;
- esecuzione delle *query* in un unico punto, per evitare duplicazione di codice, ogni metodo di SushiDB infatti richiama lo stesso metodo per eseguire la *query*.

7 Testing

7.1 Validazione

Tutte le pagine html e i file css sono stati sottoposti alla validazione tramite l'uso dei tool descritti nella sezione relativa alla realizzazione.

7.2 Peso

Il peso di ogni pagina del sito è stato calcolato tramite lo strumento descritto nella sezione relativa alla realizzazione. Tutte le pagine eccetto la galleria (3.4 MB) non superano 1 MB di peso, non è stato possibile alleggerire il peso della galleria senza perdita di qualità delle immagini.

7.3 Browser e dispositivi

Il sito è stato testato direttamente sui seguenti sistemi operativi:

- Windows 10;
- Windows 11;
- Manjaro Linux;
- IOS 15;
- Android 10;
- Android 11.

Il sito è stato testato direttamente sui seguenti browser e versioni successive:

- Microsoft Edge;
- Chrome;
- Firefox;
- Safari.

8 Organizzazione del gruppo

8.1 Processo di sviluppo

L'idea nasce dalla passione comune per il *sushi*, dopo aver fissato le funzionalità di base rispettando i requisiti imposti, siamo passati alla definizione della base di dati che regge il sito.

Successivamente è iniziata la definizione delle pagine, suddividendo il lavoro come descritto nella sezione sottostante. Dopo aver impostato una struttura e *layout* generale in del sito abbiamo creato il file *SushiDB* per interfacciarci alla base di dati e iniziare l'implementazione delle funzionalità; contemporaneamente a ciò, è maturato lo stile grafico del sito.

Dopo aver completato le funzionalità è iniziata la fase di *testing* per sistemare i possibili malfunzionamenti, fino ad arrivare al prodotto completo.

Questo processo si è svolto con riunioni a cadenza settimanale, nelle quali si sono discussi gli avanzamenti e gli obiettivi a venire, ogni tre settimane inoltre è stato fatto un *refactoring* del codice per facilitare la manutenibilità e lo sviluppo futuro.

8.2 Divisione del lavoro

8.2.1 Diego Stafa

- definizione delle pagine, comprese dei relativi file *HTML*, *CSS*, *JavaScript* e *PHP* di: *Home*, *News*, *News admin*, *Contatti*, *Info*, *Modifica profilo*, *Reimposta password*;
- definizione dei file *PHP*: *SushiDB*, *Errore 500*, *validation.php*;
- stile del sito.

8.2.2 Luciano Wu

- definizione delle pagine, comprese dei relativi file *HTML*, *CSS*, *JavaScript* e *PHP* di: *Menù*, *Menù admin*;
- definizione della base di dati;
- popolazione della base di dati;
- controlli di accessibilità.

8.2.3 Samuele Rizzato

- definizione delle pagine, comprese dei relativi file *HTML*, *CSS*, *JavaScript* e *PHP* di: *Carrello*, *Area Personale*, *Login*, *Registrazione*;
- definizione dei file *PHP*: *Global*, *Logout*, *Errore 404*;
- validazione lato *client*.
- definizione del file *.htaccess*