

UNIVERSITÀ DEGLI STUDI DI PADOVA

LAUREA IN INFORMATICA

CORSO DI TECNOLOGIE WEB

PROGETTO DI FINE CORSO

2FORCHETTE

Studente:

Luca ALESSIO

Matricola:

1070690

Studente:

Carlo SINDICO

Matricola:

1069322

INFORMAZIONI SUL SITO:

//tecnologie-web.studenti.math.unipd.it/tecweb/~csindico/

LOGIN ADMIN:

USERNAME: *admin*

PASSWORD: *admin*

Indice

1	Abstract	1
2	Utenti destinatari	2
3	Materiale consegnato	2
4	HTML	3
5	Perl	3
6	XML	6
7	Javascript	6
8	Step by step: ciclo di vita di una ricetta	8
9	CSS	9
10	Accessibilità	12
10.1	Separazione tra struttura,presentazione e comportamento . . .	12
10.2	Colori	12
10.3	Tag meta	14
10.4	Screen reader	14
11	Usabilità	14

1 Abstract

- 2Forchette.it è un sito dedicato alla raccolta di ricette culinarie.
- L'utenza che accede al sito ha la possibilità di visualizzare le ricette attualmente presenti ed eventualmente di proporle di nuove. Le ricette vengono suddivise in diverse categorie per facilitarne la consultazione e contengono dettagliate informazioni sulla loro preparazione.
- Per ogni ricetta vengono indicati una breve descrizione, la lista degli ingredienti, un'immagine rappresentativa, una spiegazione dettagliata del procedimento da seguire ed altri dati minori (quali tempo di preparazione, difficoltà, autore e numero di persone).
- La principale feature del sito è costituita dalla sezione "Proponi ricetta", dove appunto l'utente ha la possibilità di inserire una propria ricetta che verrà in seguito valutata dall'amministratore ed eventualmente ammessa nel sito.
- All'interno del sito l'utente ha inoltre le possibilità di ricercare ricette per nome (funzionalità disponibile in tutte le pagine tramite la barra di ricerca sull' header) e di lasciare eventuali commenti sul sito nella pagina dei contatti.
- Si è scelto di rendere la navigazione del sito più immediata possibile, ogni ricetta può essere infatti raggiunta con un massimo di 3 click (home \Rightarrow categoria \Rightarrow ricetta, ricerca \Rightarrow ricetta) e il design del sito è minimale e responsive per garantire la massima usabilità.
- Per quanto riguarda la parte amministrativa, l'amministratore può accedere alla propria console dal link posto nel footer della home (login: admin password: admin), da questa area riservata sarà possibile la gestione delle ricette proposte dagli utenti (deciderne quindi l'accettazione o il rifiuto) e l'eventuale eliminazione di ricette ritenute obsolete o di commenti offensivi. Queste funzionalità sono presentate in modo semplice e correlate da una breve spiegazione.

2 Utenti destinatari

- Il target di riferimento di questo sito è molto ampio, un sito di cucina viene visitato solitamente da persone appartenenti a varie fasce d'età e con differenti bagagli culturali perciò è stato scelto di rendere il sito il più semplice ed intuitivo possibile, in particolare è stata posta molta attenzione all'accessibilità per utenti non vedenti (ma questo verrà approfondito in una sezione apposita).

3 Materiale consegnato

I file consegnati sono organizzati su 3 cartelle:

- cgi-bin: cartella nella quale sono presenti i file .cgi
- data: in questa cartella sono contenuti i file xml ed i relativi XMLSchema.
- public-html: contiene index.html e le sotto-cartelle:
 - css: cartella contenente i file .css;
 - images: cartella contenente tutte le foto del sito;
 - js: cartella contenente i vari script realizzati in JavaScript.

Andiamo ora ad esaminare nel dettaglio le applicazioni di ciascuna tecnologia impiegata nel sito e le loro interazioni.

4 HTML

La struttura HTML del sito viene interamente "stampata" da vari file .cgi: questa scelta è dovuta alla grande presenza di contenuto dinamico all'interno del sito per cui non sono presenti file .html "puri" se non index.html che è però un semplice redirect a menu.cgi, file Perl che effettivamente stampa la homepage.

Le pagine HTML del sito (stampate dal cgi) sono state realizzate rispettando lo standard XHTML 1.0 Strict.

5 Perl

Le pagine scritte in Perl si dividono principalmente in due tipologie: pagine "dinamiche" di rappresentazione e pagine di elaborazione dei dati.

Alla prima tipologia appartengono i file .cgi che eseguono il "print" della pagina html con il contenuto richiesto (ne è un esempio la pagina page_template.cgi che si occupa della stampa a video di ogni ricetta) mentre le pagine della seconda tipologia sono solitamente "pagine di servizio" ovvero codice che esegue operazioni dietro le quinte come salvataggi ed eliminazioni di dati sui file xml.

Nel dettaglio, questo è ciò di cui ciascun file si occupa:

- menu.cgi - pagina principale del sito, contiene del contenuto dinamico in quanto le "ricette consigliate" che compaiono nella pagina sono casuali e variano ad ogni accesso alla pagina (inoltre in ogni pagina del sito il footer varia leggermente a seconda se l'amministratore sia loggato oppure meno).
- antipasti.cgi, secondi.cgi, primo.cgi, dessert.cgi - semplici sottomenù interni che si occupano di raggruppare le ricette appartenenti alla stessa categoria per facilitare la navigazione dell'utente
- page_template.cgi - pagina dinamica che visualizza la ricetta richiesta, funziona nel seguente modo: il link cliccato dall'utente che lo indirizza a questa pagina contiene un parametro (id) differente per ogni ricetta, page_template.cgi cerca il parametro passato tra gli attributi

IDCode delle singole ricette nel file 4forchette.xml (vedi sezione XML) ed identifica quale era la ricetta richiesta di cui va poi a restituirne le informazioni su schermo.

- `proponiricetta.cgi` - stampa la pagina "Proponi una ricetta" attraverso la quale l'utente ha la possibilità di proporre nuove ricette da inserire nel sito. La pagina contiene un form di discrete dimensioni e delle istruzioni sul suo completamento. Un'analisi più dettagliata su come avviene il processo di creazione, memorizzazione e approvazione delle ricette verrà effettuata in un successivo paragrafo.
- `handle_proposta.cgi` - quando l'utente ha compilato correttamente (vedi sezione Javascript) il form per la proposta di una nuova ricetta, la ricetta viene salvata nel "database" xml. La nuova ricetta non verrà tuttavia ancora visualizzata nel sito, prima dovrà infatti venire approvata dall'amministratore (vedi `accept_ricetta.cgi`).
- `contatti.cgi` - stampa la pagina dei contatti, qui l'utente ha la possibilità di vedere i messaggi lasciati da altri utenti ed eventualmente può lasciarne uno proprio compilando il form apposito dove viene richiesto un nome ed il testo effettivo del commento (sarebbe stato interessante implementare un sistema di registrazione e gestione degli utenti nel sito ma date le scarse dimensioni del gruppo e il tempo disponibile si è optato per una soluzione meno impegnativa).
- `inserisci_commento.cgi` - gestisce l'inserimento di un nuovo commento traducendo l'input dell'utente in dati xml.
- `amministratore_login.cgi` - contiene il form che l'amministratore utilizza per effettuare l'accesso all'area riservata. Questa pagina è accessibile solamente dal link posto nel footer della pagina principale (`menu.cgi`).
- `controllo_login.cgi` - controlla la correttezza delle credenziali inserite nel form del punto precedente. Al momento è previsto un solo amministratore per il sistema. L'amministratore viene identificato come autenticato quando un parametro apposito (`$auth`) è impostato ad un preciso valore e ciò viene settato in degli appositi cookie (vedi codice del file per i dettagli) per rendere l'informazione nota tra le diverse pagine.

- `console_admin.cgi` - stampa l'hub amministrativo da cui é possibile gestire semplicemente l'intero sito. Da questa pagina l'amministratore ha la possibilità di accettare o rifiutare le nuove ricette, eliminare vecchie ricette ed eliminare eventuali commenti ritenuti offensivi.
- `accept_ricetta.cgi` - concede ad una ricetta il permesso di venire visualizzata nel sito (semplicemente il suo attributo `accepted` viene settato positivamente).
- `delete_ricetta.cgi` - elimina una ricetta dal file `xml`. Operazione irreversibile.
- `delete_commento.cgi` - elimina un commento dal file `xml`. Operazione irreversibile.
- `logout.cgi` - breve script che effettua il logout dell'amministratore. Viene invocato alla pressione del tasto "logout" visibile sul footer di ogni pagina qualora l'amministratore sia attualmente autenticato nel sistema.
- `cercaricetta.cgi` - nell'header di ogni pagina presente una barra di ricerca, qualora l'utente digitasse una stringa di caratteri e ne richiedesse l'elaborazione, questo è lo script che verrebbe eseguito: `cercaricetta.cgi` stampa una pagina contenente un elenco di tutte le ricette il cui titolo matcha in parte o totalmente il parametro di ricerca inserito dall'utente. La ricerca avviene semplicemente scorrendo l'`xml` isolandone nomi ed id di tutte le ricette e restituendo solamente quelle che contengono una corrispondenza.
- `funzioni.pl` contiene funzioni di servizio usate in diversi contesti (ad es. `trim` degli input).

6 XML

Sono presenti 3 file xml principali con i rispettivi xsd:

- amministratore.xml - semplice file ausiliario per il controllo del corretto login nell'area amministrativa, contiene solamente due campi per il login e la password.
- commenti.xml - contiene i commenti contenuti nella sezione contatti; dalla console amministrativa possibile intervenire indirettamente sul file attuando operazioni di eliminazione
- 4forchette.xml - cuore centrale del sistema, questo documento xml svolge la funzione di database per le ricette: anche qui l'amministratore pu effettuare operazioni di eliminazione mentre l'inserimento riservato all'utente (questo aspetto è esaminato nel dettaglio nelle sezioni successive)

7 Javascript

- Javascript é stato utilizzato principalmente per il controllo degli input dei form.
- Nel sito sono presenti 3 differenti form, quello per il login nell'area amministrativa, un secondo per il submit dei commenti nella pagina dei contatti ed infine il più complesso nella sezione per la proposta di nuove ricette.
- Per quanto riguarda i primi due form (gestiti rispettivamente dai file controllo_login.js e valida_commento.js), una volta che l'utente clicca su "Submit" vengono analizzati in ordine di apparizione tutti i campi del form, se ne viene identificato uno vuoto allora l'invio viene bloccato e l'utente ne viene notificato da un messaggio d'errore specifico per quel campo.
- Per quanto riguarda il form "Proponi ricetta" invece (gestito dal file proponi_ricetta.js), oltre a questi controlli basilari ne viene effettuato un altro più approfondito nell'area di inserimento degli ingredienti.

- Dato che ogni ricetta presenta un numero di ingredienti sempre differente era impossibile prevedere un numero definito di campi per ciascun ingrediente nel form (in quanto sarebbero stati spesso insufficienti oppure troppo numerosi e comunque sgradevoli visivamente), per risolvere questo problema si è inizialmente pensato di realizzare un form dinamico che aggiungesse campi su richiesta dell'utente. Questa soluzione è stata per scartata per difficoltà tecniche nella realizzazione e principalmente per il fatto che non fosse accessibile agli screen reader.
- Si è optato quindi per lasciare la zona di inserimento degli ingredienti come una semplice area di testo all'interno della quale però vengono applicate delle regole per l'identificazione dei singoli ingredienti: dopo ogni ingrediente è necessario inserire il carattere di separazione ”;” (punto e virgola senza virgolette) ed andare a capo. Se il testo inserito dall'utente rispetta questa sintassi l'input verrà accettato, altrimenti verrà fornito un messaggio d'errore. Nella pagina sono presenti delle istruzioni sull'utilizzo del form che spiegano anche l'appropriato inserimento degli ingredienti; riteniamo che la maggior parte degli utenti non trovi difficoltà ad usare questa sintassi che è inoltre pienamente accessibile ed utilizzabile da utenti non vedenti.

Come interagiscono le varie tecnologie tra di loro? Vediamo un esempio...

8 Step by step: ciclo di vita di una ricetta

1. L'utente è intenzionato a proporre una nuova ricetta e si reca sulla pagina apposita che viene visualizzata dal file `proponiricetta.cgi`
2. L'utente ha terminato di compilare i campi e preme su "Submit"
3. Viene eseguito lo script `proponi_ricetta.js` che controlla prima di tutto se non ci sono campi vuoti e secondariamente se la sintassi del campo "Ingredienti" è corretta: se entrambe le condizioni sono soddisfatte si passa al punto successivo, altrimenti si ritorna al punto 2
4. Arrivati a questo punto siamo certi della correttezza dell'input che viene successivamente processato dal file `handle_proposta.cgi` e memorizzato in `4forchette.xml` rispettando i criteri stabiliti. In questo momento l'attributo `accepted` della nuova ricetta è impostato a 0 (che simboleggia lo stato di attesa, per cui la ricetta non verrà visualizzata all'interno del sito)
5. Al suo successivo accesso nell'area amministrativa, l'admin vedrà comparire la nuova ricetta proposta dall'utente nella sezione "Ricette proposte" da dove potrà vederne un'anteprima e di conseguenza decidere se accettarla o meno tramite gli appositi pulsanti
6. Se la ricetta è stata scartata essa verrà semplicemente eliminata dall'`xml`, se invece è stata accettata il suo attributo `accepted` verrà settato ad 1 (che simboleggia lo stato di accettazione) e la ricetta diverrà visibile all'utenza del sito.
7. In qualsiasi momento l'amministratore ha la possibilità di eliminare le ricette, attenzione però che l'eliminazione è un'operazione irreversibile!

9 CSS

- Nella realizzazione dell'interfaccia grafica del sito è stato usato lo standard CSS3.
- Allo stesso tempo si è fatta molta attenzione alla compatibilità con browser più datati, e si è cercato di utilizzare un numero ristretto delle nuove funzionalità offerte dallo standard.
- Alcune delle funzionalità CSS3 che sono state utilizzate: Border-radius e Box shadow: per realizzare i pulsanti delle form , e per le immagini.
- Nella cartella public-html/css sono presenti i seguenti fogli di stile:

style.css: modella lo stile di visualizzazione del sito sia per gli utenti desktop (che hanno uno schermo largo al massimo 1200px) che per gli utenti mobile con dispositivi con schermo piccolo (che hanno uno schermo largo massimo 390px) e dispositivi con dimensioni schermo intermedie (max-width=730px).

print.css: modella lo stile di stampa delle pagine del sito.(particolare attenzione si è data alla stampa delle pagine delle ricette).

Di seguito sono riportate le rispettive visualizzazioni del sito della versione mobile e desktop:

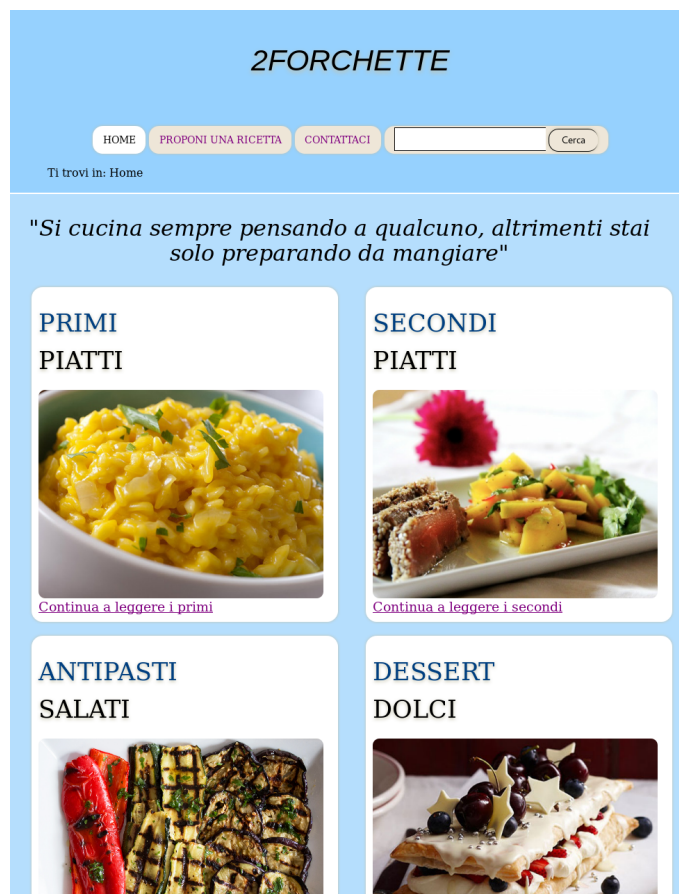


Figure 1: vedi file desktop.png



Figure 2: vedi file mobile.png

10 Accessibilità

10.1 Separazione tra struttura, presentazione e comportamento

- Per una maggiore accessibilità al sito da parte di utenti disabili e per favorire gli algoritmi dei motori di ricerca si é deciso di separare la struttura dalla presentazione e dal comportamento. Infatti il contenuto del sito é rapprentato dai file HTML e CGI, i quali richiamano i fogli di stile CSS e si utilizzano (anche in questo caso attraverso percorsi esterni),controlli in JavaScript in particolare per la compilazione dei form.
- Il contenuto rimane accessibile anche se JavaScript é disabilitato. Infatti opportuni controlli in Perl ne verificano la validità.
- Tutto il codice é stato scritto seguendo le disposizioni W3C con opportuna validazione sui loro validatori.

10.2 Colori

- Si é scelto uno schema di colori non particolarmente vivace (un mix di azzuri chiaro); anche se non sono colori di base comunque la lettura dei testi risulta accessibile. Per un sito di cucina inoltre opportuno non utilizzare colori troppi aggressivi e decisi ma attenersi ad uno stile pi sobrio e rilassato.
- I link sono sempre sottolineati, e diventano di colore viola quando vengono cliccati.

Di seguito sono riportate le visualizzazioni del sito attraverso alcuni disturbi visivi:



Figure 3: vedi file deuteranopia.png



Figure 4: vedi file protanopia.png



Figure 5: vedi file tritanopia.png

10.3 Tag meta

Sono stati inseriti per ogni pagina i tag meta:

- title: descrive la pagina corrente dal particolare al generale.
- description: da una descrizione del contenuto del sito
- keywords: contiene parole chiave per i motori di ricerca
- language: indica che il sito é stato interamente scritto in italiano.
- author: indica l'autore/i del sito
- content-type: contiene direttive per il browser
- viewport: esprime indicazioni per la visualizzazione

10.4 Screen reader

- Ogni foto ha il suo attributo alt che descrive ciò che l'immagine ritrae. Si è evitato di utilizzare immagini per visualizzare il testo, quindi il contenuto informativo rimane accessibile anche quando fallisce il caricamento delle immagini o del CSS.
- Si é fornita particolare attenzione alle parole straniere che sono state segnalate agli screenreader attraverso il tag "span xml:lang=en" segnalando la lingua con cui leggere correttamente i vocaboli.
- Inoltre è stato inserito un link skip nav per saltare direttamente al contenuto qualora l'utente dello screen reader non voglia riascoltare nuovamente il menù

11 Usabilità

L'utente riesce ad orientarsi nel sito? Analizziamo alcuni degli assi principali del giornalismo. Ometteremo gli assi Who e When dato che nel nostro contesto non è necessario dare all'utente informazioni su chi c'è dietro al sito (è un progetto didattico) nè sono presenti novità continue.

- What?: Un utente appena entra nella home capisce subito che si tratta di un sito di ricette, dalla barra dei menú (Proponi ricetta, Cerca ricetta), e dal contenuto in primo piano che mette in evidenza alcune ricette proposte.
- Where?: L'utente riesce sempre a capire dove si trova grazie al breadcrumb, inoltre l'header costituisce un importante punto di riferimento per la navigazione dell'utente.
- Why?: Perché un utente dovrebbe rimanere nel sito o dovrebbe ritornarci? Il sito é principalmente espositivo, (gli utenti possono liberamente visualizzare le ricette), si é cercato di renderlo piú interessante, aggiungendo una sezione proponi ricetta (l'utente ha la possibilità di inserire la propria ricetta), e anche una sezione commenti.
- How?: La barra di navigazione mostra tutte le sezioni principali del sito alle quali un utente può accedere. Nella barra menú é sempre evidenziata la voce della pagina in cui ci troviamo, e si vede attraverso una diversa colorazione dei link in quali altre pagine si é stati.