



Relazione Progetto Tecnologie Web

Informazioni sul gruppo

Componenti:	Alberto Nicolè 1089847 Guido Santi 1098551 Davide Tommasin 1073541 Tommaso Zagni 1070811
Referente:	Davide Tommasin 1073541 davide.tommasin.1@studenti.unipd.it

Informazioni sul sito

<http://tecnologie-web.studenti.math.unipd.it/tecweb/~dtommasi/>

Dati login

URL:	public_html/html/formLogin.html
Username:	admin
Password:	123

1 Abstract.....	1
2 Accessibilità.....	1
2.1 Separazione tra struttura, comportamento e presentazione.....	1
2.2 Schema colori	1
2.3 Tag meta.....	1
2.4 Screen reader	1
2.5 Agevolazioni per la navigazione	2
3 Usabilità	2
4 Gerarchia dei file	3
5 Struttura.....	3
6 Presentazione	3
7 Comportamento.....	4
7.1 jquery-2.1.4.min.js	4
7.2 script.js.....	4
8 Gestione dei dati	4
8.1 XMLSchema	4
8.2 XSLT	5
9 Perl	5
9.1 Pagine visitatori.....	5
9.2 Pagine amministratori	5
10 Validazione	6
Organizzazione del Gruppo	7

1 Abstract

Il progetto si pone l'obiettivo di implementare un sito che rappresenti la facciata pubblica e privata della società DaToSaAL Arredamenti (nome inventato) la quale si occupa di arredamenti interni per parrucchieri e centri estetici. Sarà quindi possibile per un utente visitatore ottenere informazioni a carattere generale sull'attività e visionare i prodotti proposti ed utilizzati ai fini dell'arredamento estetico funzionale. Non sono quindi previste interazioni da parte dell'utente visitatore mentre è prevista una sezione privata per l'amministratore il quale potrà aggiungere, modificare ed eliminare i prodotti disponibili per le categorie. Lo sviluppo del sito si è concentrato a rispettare gli standard del W3C, separazione tra struttura, comportamento e presentazione, le regole di accessibilità e la relativa usabilità.

2 Accessibilità

2.1 Separazione tra struttura, comportamento e presentazione

Nel rispetto dell'accessibilità è presente la separazione tra struttura, presentazione e comportamento.

Le pagine statiche che rappresentano la struttura sono state realizzate in XHTML 1.0 Strict assicurando quindi l'uso di tag non deprecati, rigide conformità sintattiche e controlli rigorosi. Le pagine XHTML richiamano i fogli di stile CSS che implementano la presentazione. Il comportamento è stato implementato con Perl che ci ha permesso di realizzare pagine dinamiche a partire dalle pagine statiche grazie agli script cgi. È utilizzata anche una piccola parte di JavaScript (puro) per la parte responsive, mentre sono state utilizzate le librerie JQuery, solo in parte server, per alcuni aspetti del comportamento così da garantire l'accessibilità.

2.2 Schema colori

Lo schema a colori adottato garantisce un contrasto elevato per qualsiasi utente affetto da disabilità visiva quale il daltonismo. I pochi link presenti all'interno del sito sono sottolineati e di colore blu per facilitare tutti gli utenti, eccezione il menu di navigazione. In caso di link già visitato il testo relativo diventa di colore viola scuro.

Abbiamo utilizzato il servizio offerto dal sito <http://www.vischeck.com/> per verificare l'accessibilità visiva.

2.3 Tag meta

In ogni pagina sono presenti i seguenti tag meta: Content-Type, charset, title, author, description, keywords, languages e viewport. Come indicato dal tag languages il sito è interamente scritto in italiano ma essendo presenti alcune parole in inglese abbiamo aiutato gli screen reader tramite l'uso del tag e l'attributo xml:lang="" così da fornire una lettura corretta e precisa.

2.4 Screen reader

Oltre all'uso dei tag citato sopra, essendo il sito una presentazione di vari prodotti abbiamo curato che in tutte le immagini fossero presenti gli attributi 'alt' garantendo così la descrizione completa di ciò che ritrae.

All'interno dei tag <form> sono presenti gli attributi 'label' per associare un testo ad ogni campo della form.

Non sono state utilizzate immagini di alcun genere per rappresentare testo, garantendo così accessibilità anche in caso di mancato caricamento delle immagini o del CSS.

2.5 Agevolazioni per la navigazione

Sono state curate le seguenti agevolazioni pensate ad hoc per gli utenti con disabilità:

- Tabindex: definiti per ogni form, permettono una navigazione agevolata tramite l'uso del tasto tab che sposta l'attenzione all'input successivo.
- Link per spostarsi al contenuto: prima della nav bar è presente un link visibile solo agli screen reader per saltare direttamente al contenuto. Così da evitare la lettura dell'intera barra di navigazione.
- Link per tornare al menu: nella parte responsive è stato inserito un link subito prima del footer con scritto "torna su" che permette di tornare ad inizio pagina.

3 Usabilità

Abbiamo curato l'accessibilità cercando di seguire le raccomandazioni:

- La homepage del sito cerca di rispondere in modo esaustivo:
 - Where: essendo un sito di presentazione/illustrazione il testo nella home spiega l'importanza dell'avere un arredo giusto nelle sale d'acconciatori/estetiche. Punto chiave è dimostrare all'utente visualizzatore che l'azienda è capace risolvere tutti i problemi d'arredo.
 - Who: al primo accesso l'utente deve capire con che entità si trova ad interagire, è stato inserito il logo aziendale in alto a sinistra seguito dal nome. Ormai è luogo comune che il logo (presente in ogni pagina) sia cliccabile, così è stato fatto mantenendo il consueto reindirizzamento alla home.
 - Why: i benefici del sito sono espressi nel testo iniziale che tenta di convincere l'utente a scegliere l'azienda per arredare il suo locale.
 - When: già dall'home grazie alla sezione prodotti in vista, inserita subito dopo la presentazione, viene catturata l'attenzione dell'utente visitatore sulle vasta tipologia di prodotto utilizzati dall'azienda.
 - What: grazie al menù è possibile focalizzare la propria attenzione sulle due categorie 'Acconciatori' ed 'Estetiche' che andranno a illustrare la lista dei prodotti, divisa in sezioni per ognuna. Riteniamo sia quindi facilmente intuibile reperire le informazioni dal sito.
 - How: La barra di navigazione illustra tutte le sezioni visitabili da un utente non autenticato che quindi può accedere solamente alle sezioni illustrative.
- Navbar: nella barra di navigazione viene evidenziata la pagina visitata, dove è disabilitato il click evitando così un inutile refresh. Le altre voci del menù invece se puntate col mouse indicano la possibilità di navigazione verso un'altra pagina.
- Breadcrumbs: per facilitare l'utente a riconoscere la sezione visitata è stato implementato il classico breadcrumbs per riconoscere il percorso effettuato. Esso è situato sotto la navbar. Anche se attualmente il sito non presenta grandi annidamenti questo può comunque essere utile per una futura espansione.

4 Gerarchia dei file

Abbiamo organizzato i file del sito secondo questa gerarchia:

- public_html:
 - html: contiene tutte le pagine statiche del sito.
 - css: che contiene i file css per la presentazione.
 - images: contiene tutte le immagini presenti del sito, sia quelle statiche che quelle caricate dall'utente amministratore.
 - js: contenente i file JavaScript e la libreria jQuery.
- cgi-bin: contiene tutti gli script cgi realizzati con perl.
- data: contiene tutti i file xml, schemi e istanze.

5 Struttura

all'interno della cartella public_html/html si trovano i vari file delle pagine statiche .html.

Tutte le pagine del progetto sono state realizzate secondo lo standard XHTML 1.0 Strict.

le pagine statiche che costituiscono il sito sono:

- index.html: Questa pagina ha il solo scopo di reindirizzare l'utente verso la vera home del sito cgi-bin/index.cgi
- chi-siamo.html: semplice descrizione dell'azienda e della sua storia.
- contatti: tutte le informazioni basilari come, dove contattare e trovare l'azienda.
- formLogin: il form che permette di inserire i dati i quali verranno verificati dall'apposita pagina .cgi e successivamente, se i dati sono corretti, reindirizzato alla home backend.

6 Presentazione

Abbiamo utilizzato lo standard CSS3 per realizzare l'interfaccia grafica, cercando di utilizzare il meno possibile funzionalità recenti, in modo di garantire una buona compatibilità anche con browser datati. I file CSS sono presenti nella cartella public_html/css e sono organizzati in:

- main.css: css principale che gestisce tutte le pagine in modalità desktop
- responsive.css: il quale andrà a sovrascrivere determinate caratteristiche del main.css in modo da adattare il sito al nostro device
- print.css: utilizzato per avere una semplice stampa su carta, eliminando la grafica in eccesso

7 Comportamento

Per rendere il sito facile da usare abbiamo utilizzato la tecnologia Javascript, l'abbiamo usata soprattutto per avere un buon utilizzo nella parte per i dispositivi mobile. Nello specifico abbiamo utilizzato la libreria jQuery di Javascript per poter modificare le pagine html e css dinamicamente, così da rendere il sito più user friendly.

7.1 jquery-2.1.4.min.js

jquery-2.1.4.min.js non è nient'altro che la libreria jQuery da importare per poter usare le varie funzioni.

7.2 script.js

Script.js è la pagina Javascript che contiene tre funzioni, create appositamente per il sito in versione mobile, così da migliorare l'esperienza di utilizzo.

- \$(".btn-responsive-menu").click(function(): è la funzione che ci permette, quando siamo in versione mobile, di poter schiacciare sopra al pulsante del menù, così da aprire il menu e poter cambiare eventualmente pagina. Questa funzione aggiunge due classi in modo da far cambiare il css agli elementi dove vengono aggiunti le classi.
- \$(window).resize(function(): questa funzione invece fa esattamente l'opposto, se si ridimensiona la pagina mentre si ha il menu aperto (da responsive) toglie le due classi create precedentemente dalla funzione \$(".btn-responsive-menu").click(function() così da ripristinare in css per il sito in versione desktop.
- ("freccia").click(function(): funzione che quando si schiaccia nell'apposita scritta per tornare all'inizio pagina, invece che far sparire la pagina e ricomparire all'inizio fa lo scroll in velocità della pagine, una funzione puramente estetica.

8 Gestione dei dati

L'utente autorizzato può gestire l'inserimento, la modifica e la cancellazione dei prodotti. Essendo il sito destinato per visualizzare prodotti in vendita, la parte backend è stata nascosta agli occhi degli utenti finali, per cui senza mettere un pulsante "login" all'interno del sito, solo l'amministratore sarà dotato del URL con relativo username e password per accedere alla parte di backend.

Abbiamo utilizzato file xml per definire schemi in XMLSchema e le relative istanze.

8.1 XMLSchema

La scelta di utilizzare XMLSchema anziché DTD è dettata dal fatto che è possibile specificare più vincoli su elementi e attributi. XMLSchema risulta più pratico ed espressivo sebbene siano necessarie più righe di codice. Tutti gli schemi seguono il più possibile il modello "tende alla veneziana" dove tutti i tipi sono nominali e tutti gli elementi sono locali.

8.2 XSLT

La tecnologia XSLT è stata usata nelle prime fasi preliminari di creazione per realizzare dei template che mostrassero in output html l'esistenza dei dati che venivano testati. La scelta di abbandonare tale tecnologia in favore di una gestione completa dei dati in Perl ci ha reso il lavoro più semplice.

9 Perl

La visualizzazione delle informazioni dinamiche avviene tramite Perl che inserisce i dati all'interno delle pagine html. Gli script CGI sono applicati sia per utenti amministratori sia per utenti visitatori.

La gestione dei dati XML avviene tramite il modulo LibXML che permette di leggere e scrivere un file XML tramite Perl. Per l'estrazione dei dati abbiamo usato espressioni XPath.

9.1 Pagine visitatori

I visitatori possono visualizzare nella home i prodotti in vista tramite lo script index.cgi che preleva i prodotti (chiamati oggetti del file nostroCatalogo.xml) con l'attributo in_vista="true" e si occupa di mostrarli a schermo fornendo alcune informazioni.

All'interno del menù è possibile accedere alle sezioni "Acconciatori" e "Estetiche", tramite gli script omonimi. Questi script realizzano pagine html divise in tipologie di articoli ognuna delle quali mostra degli esempi (da 0 a 3) per ogni tipo di categoria. Cliccando sul tipo di categoria interessata viene lanciato lo script sottocategoria.cgi che si occupa di caricare lista d'oggetti appartenenti al tipo d'oggetto.

Infine lo script print-oggetto.cgi stampa a schermo tutte le informazioni relative all'oggetto richiesto.

9.2 Pagine amministratori

La gestione degli accessi alla parte privata del sito viene gestita tramite lo script login.cgi che si occupa di controllare la validità dei dati inseriti basandosi sul file utenti.xml. Tale script rimanderà poi alla homepage del back-end.

Scorrendo il menù del back-end l'amministratore troverà varie funzionalità utili alla gestione del sito. Gli script principali sono:

- listing-prodotti.cgi si occupa di mostrare tutti gli oggetti esistenti all'interno del file nostroCatalogo.xml mostrando id, nome e dando la possibilità di modificare o rimuovere gli oggetti.
- La modifica degli oggetti richiede due script:
 - formModificaOggetto.cgi mostra all'interno di una pagina html le informazioni relative a quel particolare oggetto, dando la possibilità all'amministratore di modificarle.
 - modifica-oggetto.cgi raccoglie le informazioni fornite da formModificaOggetto.cgi e modifica l'oggetto presente nel file xml nostraCatalogo.xml.
- aggiungi-oggetto.cgi prende le informazioni fornite da nuovo_oggetto.html e aggiunge un nuovo oggetto nel file nostroCatalogo.xml
- nota: le pagine aggiungi-oggetto.cgi e modifica-oggetto.ci utilizzano la libreria HTML::Entities per permettere l'encoding dell'input, evitando così un inserimento errato da parte dell'utente.

10 Validazione

Per validare il codice html abbiamo usato il validatore presente sul web all'indirizzo <https://validator.w3.org>.

Per validare il codice css abbiamo usato il validatore presente sul web all'indirizzo <https://jigsaw.w3.org/css-validator> .

Per validare i file XML è stato usato un pratico validatore presente sul web all'indirizzo <http://www.utilities-online.info/xsdvalidation> che tra le varie funzionalità offre la validazione di un file XML rispetto ad un XMLSchema e inoltre segnala se il file su cui si sta lavorando è ben formato o meno.

Organizzazione del Gruppo

Il lavoro su questo progetto è stato così distribuito:

- Alberto Nicolè:
 - Lavoro su file .html
 - Lavoro sulla parte Perl
 - Lavoro su file .css
 - Validazione pagine
- Guido Santi:
 - Lavoro sulla parte Perl
 - Lavoro su file .html
 - Validazione pagine
- Davide Tommasin:
 - Lavoro su file .html
 - Lavoro su file .css
 - Lavoro sulla parte Javascript
 - Stesura della relazione
- Tommaso Zagni:
 - Lavoro su file .html
 - Lavoro su file .css
 - Lavoro sulla parte XML
 - Stesura della relazione