



federico.rispo@studenti.unipd.it

Relazione di Progetto

Informazioni sul Documento

Referente	Federico Rispo
Autori	Federico Rispo 594374 Federico Brian 1122698 Georgiana Uglea 1121248 Alessandro Canesso 1122701
Lista distribuzione	<i>Prof. Lamberto Ballan</i> <i>Prof. Ombretta Gaggi</i> <i>Dott. Matteo Ciman</i>

Scopo del Documento

*Il presente documento contiene la relazione del progetto didattico
del corso di Tecnologie Web svolto nell'A.A. 2018/2019*

Indirizzo del sito web:

<http://tecweb1819.studenti.math.unipd.it/frispo>

Abstract

Il progetto, di nome “Usauto” ha come obiettivo la creazione di un sito web dedicato alla rivendita di automobili usate. Lo scopo del sito è di permettere agli utenti di accordarsi per la vendita o l’acquisto di mezzi pubblicati su un mercato comune. I visitatori del sito possono visualizzare il mercato e cercare mezzi con qualità specifiche, per sbloccare la totalità delle funzioni del sito è però necessaria la registrazione, la quale permette di diventare utenti. Una volta effettuata la registrazione, agli utenti è permesso pubblicare annunci riguardanti mezzi che hanno intenzione di vendere oppure contattare altri utenti per acquistare o sapere di più su automobili da loro messe sul mercato.

Indice

1	Introduzione	1
1.1	Analisi d'utenza	1
1.1.1	Guidatori inesperti	1
1.1.2	Guidatori esperti	1
1.2	Considerazioni finali	1
2	Sviluppo	2
2.1	Progettazione	2
2.2	Design	2
2.3	HTML e CSS	2
2.4	MySQL e PHP	3
2.5	JavaScript	3
3	Accessibilità	5
3.1	Separazione tra contenuto, presentazione e struttura	5
3.2	Facilitazioni per la navigazione	5
3.3	Schema colori	6
3.4	Accessibilità mobile	6
3.5	CSS di stampa	7
4	Installazione	8
4.1	Requisiti	8
4.2	Configurazione e dump	8
5	Organizzazione interna	9
5.1	Suddivisione interna dei compiti	9
6	Tecnologie utilizzate	10
6.1	Elenco strumenti	10

Elenco delle figure

1	Schema dei colori	6
---	-----------------------------	---

Elenco delle tabelle

1	Elementi implementati nel JavaScript	4
---	--	---

1 Introduzione

1.1 Analisi d'utenza

Il sito è rivolto ad utenti in età adulta, intenzionati a vendere un'automobile di loro proprietà od in cerca di un mezzo.

1.1.1 Guidatori inesperti

I guidatori inesperti sono un tipo di utenza giovane, intesi come i neopatentati ed i guidatori che hanno ottenuto la patente di guida da meno di dieci anni.

Abituati alla tecnologia, sono alla ricerca di un sito web moderno per poter acquistare la loro prima auto, eseguendo ricerche veloci preferibilmente dal loro dispositivo mobile.

1.1.2 Guidatori esperti

I Guidatori esperti sono un tipo di utenza tendenzialmente poco avvezza alle tecnologie, poiché si riferisce alla fascia d'età delle persone con età maggiore ai 28 anni. Essi comprendono lavoratori e/o pensionati in cerca di un'auto per sé e padri di famiglia in cerca, ad esempio, della prima auto da acquistare o regalare al proprio figlio neopatentato. Costoro effettueranno ricerche ad ampio raggio, poiché vorranno visionare tutte le auto appartenenti ad una determinata classe (per stile, cilindrata, ...) prima di decidere quale acquistare.

1.2 Considerazioni finali

Attraverso un'interfaccia intuitiva ed un linguaggio informale il sito cerca di essere fruibile alla quasi totalità delle persone interessate alla compravendita di automobili, inoltre si cerca di non allontanare i meno esperti permettendo una ricerca modulare con molti campi opzionali. In questa maniera si spera di non intimidire gli utenti con poca dimestichezza per quanto riguarda le componenti specifiche delle automobili. Questo vale anche per la sezione annunci: è infatti possibile pubblicare un annuncio senza scendere nei dettagli riguardanti le qualità di un veicolo.

2 Sviluppo

2.1 Progettazione

La tipologia del sito è stata decisa durante la fase preliminare: in seguito ad un *brainstorming* si sono delineate le idee del gruppo; idee che hanno permesso di elencare gli obiettivi da raggiungere durante lo sviluppo del progetto.

Successivamente, è stata organizzata una suddivisione generale delle responsabilità individuali.

Una volta fatti i primi passi si sono decisi il layout del sito, i pattern da seguire per lo sviluppo, le porzioni principali del sito e le tecnologie da utilizzare.

Infine, è stato progettato il database, il quale permette l'immagazzinamento e l'organizzazione delle informazioni sulle quali è fondato il sito web: **inserzioni**, **utenti**, **province**, **regioni**, **equipaggiamenti** montati dalle macchine, vari **optional** delle macchine, le **macchine** stesse, le **categorie** a cui appartengono le macchine e i **colori** delle macchine. Si è passati poi allo sviluppo del front-end e del back-end.

Dopo aver effettuato il merge del codice, il sito è stato collaudato per risolvere eventuali bug e verificare l'accessibilità e la compatibilità con dispositivi e browser differenti.

2.2 Design

Dato lo stato in continuo aggiornamento del sito, è stata posta particolare attenzione alla separazione tra contenuto statico e dinamico e alla separazione tra struttura, presentazione e comportamento.

Come standard si è deciso di usare **XHTML 1.0 STRICT**, il quale fornisce un ottimo supporto per strutturare il contenuto in maniera semplice ed efficace: ad esempio, grazie alla sua sintassi rigida, i validatori possono rilevare immediatamente la chiusura corretta dei tag. Le strutture delle pagine web sono state “templatizzate” in modo da incrementare la robustezza del sito web, sul quale sono applicati fogli di stile in CSS puri. I layout in CSS sono stati sviluppati con l'obiettivo di ottenere un design fluido e funzionale per la maggior parte dei browsers e dispositivi. Inoltre attraverso *media queries* si sono resi disponibili diversi layout per finestre browser di diverse risoluzioni, in fine Il supporto a mobile e tablet è stato fornito tramite appositi fogli di stile. Un altro aspetto preso in considerazione è stata l'accessibilità del sito per diverse categorie di utenti, questa è analizzata in dettaglio maggiore successivamente, in una sezione dedicata.

2.3 HTML e CSS

Riuscire a separare la struttura delle informazioni dallo stile è stato fondamentale per il tipo di sito preso in esame.

È stato utilizzato un solo file CSS, opportunamente diviso in categorie, per tutte le direttive di design esposte nelle varie pagine. Nello stesso file troviamo anche

le direttive specifiche per il sito in modalità mobile e tablet. Invece si è scelto di separare il css relativo alla stampa.

Per quanto riguarda le grandezze si è fatta molta attenzione ad esprimerle in unità relative (**rem**, **em**, %).

2.4 MySQL e PHP

Il lato server fornisce un supporto fondamentale a tutto ciò che succede nel sito web, il database è stato sviluppato per registrare utenti e annunci, con una grande quantità di dettagli annessi. (Aggiungere)

L'altra componente cruciale server side è PHP, che gestisce tutte le richieste ricevute dal server effettuando redirect alle componenti assegnate per soddisfare le richieste: ogni richiesta di azione o pagina viene passata attraverso index.php che ne verifica la legittimità e reindirizza al componente del controller assegnato a soddisfare tale richiesta, in caso contrario viene lanciato il rispettivo errore.

Il controller gestendo la richiesta carica i dati dal database ed elabora il template da ritornare all'utente. (Aggiungere)

Questo permette una comoda gestione delle pagine in modo dinamico e anche facili modifiche inserendo nuove pagine senza dover eliminare versioni precedenti, con modifiche minime al controller. (Aggiungere)

2.5 JavaScript

È stato utilizzato il linguaggio JavaScript per rendere dinamico il comportamento *client-side* di alcune pagine di UsAuto. In particolare, l'utilizzo di JavaScript è circoscritto al controllo dell'input nei *form* di ricerca: questo è il frutto di una decisione presa all'unanimità in quanto non è possibile fare alcuna assunzione circa l'utilizzo di tale tecnologia, poiché può essere disabilitata o non presente. Proprio per questo motivo il controllo dell'input non è eseguito solamente da funzioni JavaScript: nell'eventualità che esso non sia presente, sarà compito del **PHP** fornire un controllo *server-side*, il quale fornirà una visualizzazione opportuna in caso di errore. Di seguito sono elencate le funzioni implementate, seguite da una breve descrizione.

Elemento	Descrizione
wrongInput	Messaggio d'errore standard da visualizzare in caso di input sbagliato
letters	Espressione regolare che ammette solo lettere maiuscole, minuscole e lettere accentate
numbers	Espressione regolare che ammette solo numeri, da zero a nove
errors[]	Array che contiene la form che ha generato l'errore con il messaggio d'errore da visualizzare

– continua a pagina successiva

– <i>continuazione da pagina precedente</i>	
<code>displayError()</code>	Funzione che controlla l'array <code>errors</code> e decide se visualizzare il messaggio d'errore e disabilitare il pulsante di ricerca
<code>checkLetters()</code>	Funzione che controlla che l'input delle form sia corretto, ammettendo solo lettere
<code>checkNumbers()</code>	Funzione che controlla che l'input delle form sia corretto, ammettendo solo numeri
<code>invalidInput(element, paragraph, message)</code>	Funzione invocata in presenza di un input errato in una form. Riceve come parametri attuali l'elemento (form) contenente l'input errato, l'id del paragrafo dove visualizzare il messaggio d'errore (tipicamente <code>inputErr</code>) ed il messaggio d'errore da visualizzare. Si occupa di modificare lo stile della form per darle una colorazione del bordo rossa e ad aggiungere l'errore all'array <code>errors</code> , se esso non lo contiene già. Viene infine invocata la funzione <code>displayErrors()</code>
<code>validInput(val, element, paragraph)</code>	Funzione invocata in presenza di un input corretto in una form. Riceve come parametri attuali il valore corretto da visualizzare (tipicamente il valore già contenuto dalla form), l'elemento interessato (form) contenente l'input corretto e l'id del paragrafo che contiene gli eventuali messaggi d'errore da visualizzare (tipicamente <code>inputErr</code>). Si occupa di modificare lo stile della form per darle una colorazione del bordo nera e a togliere l'errore all'array <code>errors</code> . Viene infine invocata la funzione <code>displayErrors()</code>

Tabella 1: Elementi implementati nel JavaScript

3 Accessibilità

3.1 Separazione tra contenuto, presentazione e struttura

Per migliorare la qualità dell'utilizzo del sito agli utenti con disabilità è stato posto l'accento sulla separazione tra presentazione, struttura e comportamento. La separazione è stata implementata tramite documenti XHTML strict 1.0, che richiamano fogli di stile CSS esterni.

Nonostante il controllo input e i messaggi di errore siano stati realizzati in JavaScript, esistono funzioni PHP per le stesse operazioni che permettono il corretto funzionamento del sito anche su dispositivi che non lo supportano. La totalità del codice è stata scritta seguendo le raccomandazioni del **W3C**¹, accertandosi che fossero rispettate tramite validazione.

3.2 Facilitazioni per la navigazione

L'interfaccia del sito è stata ideata per avere una navigazione comoda attraverso tab, non è stato quindi necessario aggiungere un attributo `tabindex` per dare un ordine prioritario alle tabulazioni. Tutte le immagini sono state marcate con apposite label e tag `alt`.

In modo da poter essere accessibile da dispositivi con schermi di dimensioni minori rispetto ai desktop, il sito web è stato costruito per avere un layout *responsive*. Sempre con questo scopo è stata posta particolare attenzione allo sviluppo orizzontale del sito, evitando in qualunque circostanza lo scroll orizzontale. Questo è risultato fondamentale per l'usabilità del sito su smartphone.

Con lo scopo di minimizzare il disorientamento e la confusione dei visitatori è stata creata una pagina di **errore 404**, contenente link utili a riprendere la navigazione nel sito e un logo personalizzato che, oltre all'aggiungere particolarità alla pagina, cerca di ridurre la frustrazione dell'utente sfruttando l'ironia.

Ogni link è stato reso distinguibile da ogni altro tramite appositi CSS, **hover** e **visited**: nel caso dell'hover la sottolineatura del link si ispessisce e scurisce, nel caso del visited il link cambia colore. All'apice di ogni pagina è presente una barra di navigazione che aiuta l'utente a orientarsi nel sito, evidenziando in ogni momento il nome della pagina in cui si trova. Inoltre, avendo rispettato la regola dei 4 click, non abbiamo giudicato necessaria la creazione di una sitemap.

Un ulteriore aiuto al visitatore viene fornito dal breadcrumb. Il breadcrumb tiene traccia del percorso dell'utente nel sito rispetto alla home, consentendo di tornare con facilità alle pagine visitate in precedenza.

L'accessibilità del sito è stata migliorata ulteriormente dalla presenza di un tasto ancora che permette di tornare all'apice della pagina.

¹<https://www.w3.org/>

3.3 Schema colori

Per quello che concerne l'accessibilità delle categorie di utenti con difficoltà visive si è deciso di usare unicamente colori “**web safe**”, mantenendo sempre un contrasto elevato tra lo sfondo e le scritte, evitando così di renderle meno leggibili; non è presente inoltre alcun contenuto pericoloso per utenti a rischio epilessia, o contenuti che possano indurre l'utente in stato confusionale. Per quanto riguarda le immagini non è stato possibile fare molto perché vengono caricate da utenti, è possibile però ingrandirle cliccandoci sopra.
(Inserire screenshot del sito versione desktop anche con modalità per disabili)

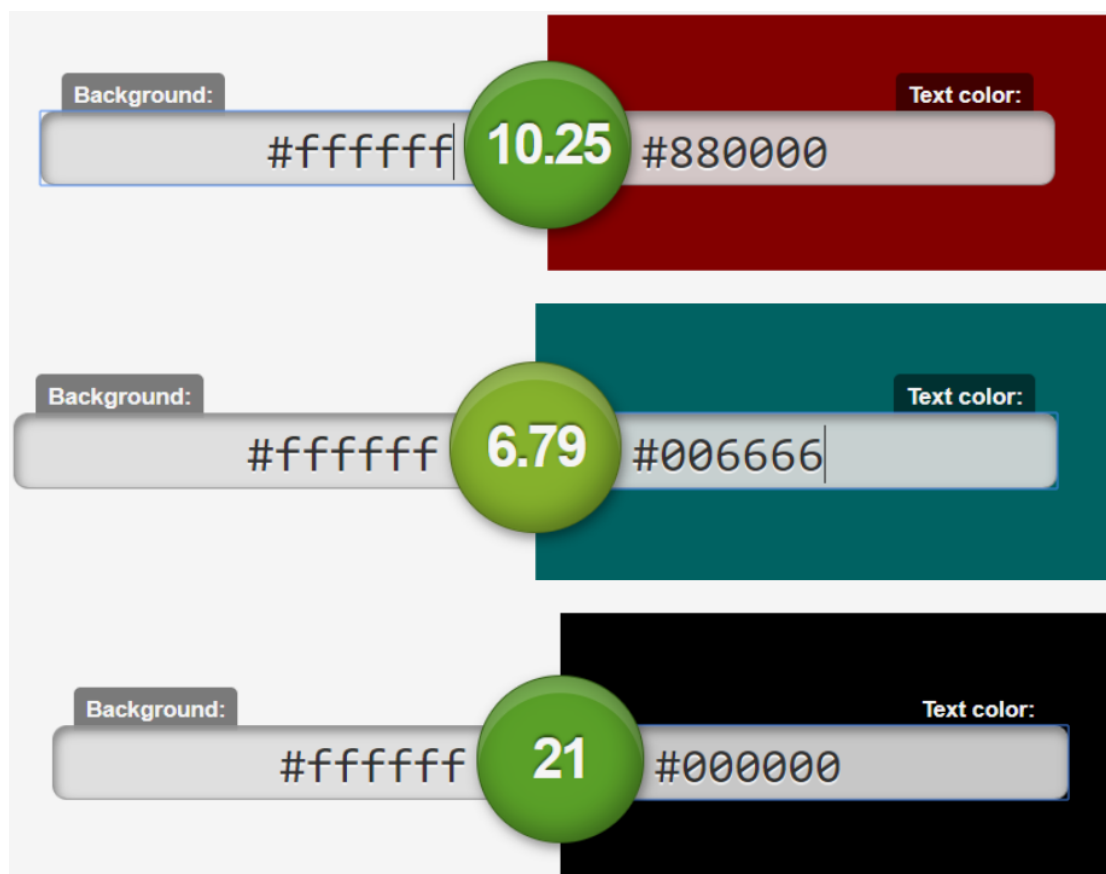


Figura 1: Schema dei colori

3.4 Accessibilità mobile

L'accessibilità per mobile è stata sviluppata con CSS puro, questo ha garantito la compatibilità con la maggior parte dei dispositivi mobile.
(Inserire screenshot del sito versione mobile)

3.5 CSS di stampa

Viene incluso un layout di stampa per ogni pagina: le informazioni principali di una pagina vengono estrapolate e organizzate in un formato adeguato.
(Inserire screenshot)

4 Installazione

4.1 Requisiti

Per un corretto funzionamento del sito i requisiti minimi sono i seguenti:

- PHP versione 7.0 o migliore;
- Un server HTTP - Apache consigliato;
- Un motore database compatibile con MySQL.

4.2 Configurazione e dump

(DA RIVEDERE)

Il primo accesso al server, per un funzionamento corretto, deve essere fatto attraverso la pagina iniziale “./php/index.php” che imposta i parametri necessari per la navigazione nel sito.

Per un funzionamento corretto sono necessarie alcune configurazioni:

- estensione “mysqli” di PHP attiva;
- estensione “PDO” attiva in `php.ini`;
- permessi di upload concessi a PHP.

Durante il dumping si consiglia, con il comando “`set foreign_key_checks=0`”, di disattivare le chiave esterne e di riattivarle una volta terminata la fase di dumping.

5 Organizzazione interna

5.1 Suddivisione interna dei compiti

- **Referente:** Federico Rispo;
- **HTML e CSS:** Georgiana Uglea, Federico Rispo;
- **MySQL:** Federico Rispo, Alessandro Canesso;
- **PHP:** Federico Rispo;
- **JavaScript:** Federico Brian;
- **Grafiche:** Georgiana Uglea;
- **Contenuti:** Alessandro Canesso, Georgiana Uglea;
- **Accessibilità:** Federico Rispo;
- **Relazione:** Alessandro Canesso, Federico Brian;
- **Testing:** Federico Brian.

6 Tecnologie utilizzate

6.1 Elenco strumenti

- **Versionamento:** Git, GitLab;
- **IDE:** Vim, Geany, WebStorm, PhpStorm;
- **Stesura relazione:** Microsoft Word, TeXMaker (per L^AT_EX);
- **Grafica:** Photoshop, Adobe Illustrator, GIMP;
- **DBMS:** MariaDB;
- **Test di contrasto dei colori:** <https://contrast-ratio.com/>
- **Verifica indice di leggibilità:**
https://farfalla-project.org/readability_static/
- **Verifica accessibilità:** <https://achecker.ca/checker/index.php>