



Adrenaline Motocross Park

Progetto del corso di Tecnologie Web
A.A. 2021/2022

Informazioni sul gruppo

Membri	Alessandro Cavaliere	1224440
	Filippo Brugnolaro	1217321
	Riccardo Simionato	1230236
	Leonardo Gambirasio	1224453

Referente

Alessandro Cavaliere
alessandro.cavaliere.1@studenti.unipd.it

Utenti

Amministratore: admin - admin
Utente generico: user - user

Indice

1	Introduzione	2
1.1	Abstract	2
2	Analisi	3
2.1	Studio dell'utenza finale	3
2.2	Possibili ricerche sui motori di ricerca	3
3	Progettazione	4
3.1	Obiettivi	4
3.2	Layout	4
3.2.1	Resize e Mobile	4
3.3	Accessibilità	4
3.3.1	Trasformazione elegante	4
3.3.2	Schema colori	4
3.4	Schema organizzativo	4
4	Implementazione	5
4.1	Linguaggi e strumenti	5
4.1.1	HTML 5	5
4.1.2	PHP	5
4.1.3	SQL	5
4.1.4	JavaScript	6
4.1.5	XAMPP	6
4.1.6	Ispezione di Chrome	6
4.2	Funzionamento generale	7
4.3	Intrusività	7
5	Validazione	8
5.1	Strumenti usati	8
5.1.1	TotalValidator	8
5.1.2	W3C HTML Validator	8
5.1.3	W3C CSS Validator	8
5.1.4	PhpCodeChecker	8
5.1.5	Esprima	8
6	Organizzazione del lavoro	9

Appendice 10

A	Note per la correzione	10
A.1	Credenziali	10
A.2	Popolamento	10

1 Introduzione

1.1 Abstract

Il progetto *Adrenaline Motocross Park*, svolto per il corso di Tecnologie Web nell'anno accademico 2021-2022, propone di implementare un sito Internet adibito a facilitare la gestione delle prenotazioni dei servizi offerti ai clienti.

Per conseguire lo scopo stabilito, si è deciso di creare qualcosa di elegante ed efficace allo stesso tempo, in modo da garantire all'utente un'esperienza piacevole. Per i visitatori del sito è possibile registrarsi e creare il proprio account personale, grazie al quale l'utente può prenotare il suo ingresso o i corsi ai quali è eventualmente interessato a partecipare.

Se un utente non volesse iscriversi, il sito offre comunque una funzionalità informativa.

2 Analisi

2.1 Studio dell'utenza finale

Il progetto *Adrenaline Motocross Park* si propone come piattaforma di prenotazione per un parco divertimenti dedicato ai motociclisti e agli appassionati di motocross. L'intervallo d'età si attesta tra i 12 e i 45 anni circa e, essendo comunque uno sport molto particolare, il linguaggio sarà nella maggior parte semplice e più tecnico in alcuni tratti (pochi). Essendo pensata più come piattaforma di prenotazione per i clienti dell'impianto, si presuppone che l'utenza conosca il linguaggio tecnico dello sport, ma abbiamo comunque pensato anche agli utenti che si vogliono affacciare per la prima volta.

2.2 Possibili ricerche sui motori di ricerca

3 Progettazione

3.1 Obiettivi

3.2 Layout

3.2.1 Resize e Mobile

3.3 Accessibilità

3.3.1 Trasformazione elegante

3.3.2 Schema colori

3.4 Schema organizzativo

4 Implementazione

4.1 Linguaggi e strumenti

In questa sezione verranno illustrati i linguaggi e gli strumenti che il gruppo ha usato durante lo sviluppo del progetto.

4.1.1 HTML 5

Il gruppo ha utilizzato il linguaggio HTML5¹ cercando di mantenere il più possibile la retrocompatibilità con XHTML², così da rendere il sito accessibile anche su dispositivi con browser più obsoleti. Inoltre la verbosità del secondo rispetto al primo ha contribuito notevolmente sulla scelta finale. Per assicurare un codice corretto, sono state seguite le linee guida del corso di Tecnologie Web. Il codice è stato validato utilizzando il tool di validazione W3C³.

In particolari si ha riposto attenzione alle seguenti peculiarità:

- **Chiusura tag:** ogni tag deve essere chiuso(<tag></tag> oppure <tag/>);
- **Metatag:** nella sezione header, devono essere inseriti i metatag necessari per migliorare l'accessibilità verso i motori di ricerca. Questo permette al sito di avere una migliore visibilità in internet. (sezione relativa alle Possibili ricerche sui motori di ricerca);
- **Separazione struttura-presentazione-comportamento:** il codice HTML non deve contenere CSS o script. Questi devono essere scritti in file separati e importati nell'header;
- **Struttura:** il codice HTML non deve sostituire il CSS.

4.1.2 PHP

4.1.3 SQL

SQL è stato utilizzato per la codifica del database. Il database è composto dalle seguenti tabelle:

- **data_disponibile:** contiene le *date* di apertura dell'impianto che vengono inserite dall'amministratore; viene inoltre stabilito il numero di *posti* (esclusi i piloti partecipanti ai corsi) disponibili in ogni data di apertura;
- **ingressi_entrata:** contiene la *data* e la licenza dell'*utente* che ha prenotato l'ingresso;
- **ingressi_lezione:** contiene la *data* e la licenza dell'*utente* che ha prenotato la lezione;
- **lezione:** contiene tutte le lezioni inserite dall'amministratore; ogni lezione contiene la *data* in cui verrà svolta, l'*istruttore* che la svolgerà in una *pista* assegnata, la descrizione e il numero di posti disponibili;

¹<https://html.spec.whatwg.org/multipage/>

²<http://www.w3.org/TR/2018/SPSD-xhtml1-20180327/>

³<https://validator.w3.org/>

- **messaggio:** contiene l'*oggetto*, il *testo* e la *data* di invio del messaggio; per identificare chi lo ha inviato si hanno *nominativo*, *email* e *telefono* dell'utente;
- **moto:** contiene le moto disponibili dell'impianto; per identificarle si ha la *marca*, il *modello*, la *cilindrata* e l'*anno* di produzione;
- **noleggio:** contiene informazioni riguardo al noleggio in una determinata *data* di *attrezzatura* o *moto* da parte di un *utente*;
- **pista:** contiene tutte le piste inserite dall'amministratore; contiene la *lunghezza* e il tipo di *terreno* della pista accompagnato da una *descrizione* e dagli orari di *apertura* e *chiusura*; è possibile, ma non indispensabile, inserire una *foto* del tracciato stesso;
- **utente:** contiene tutti le informazioni degli utenti/amministratori (in base al loro *ruolo* all'interno del sito); è dunque presente *cognome*, *nome* della persona con relativa data di *nascita*, codice fiscale (*cf*) e numero di *telefono*; vengono memorizzate anche le credenziali come l'*email* e la *password*;

4.1.4 JavaScript

Il linguaggio Javascript è stato utilizzato per due scopi principalmente:

1. per convalidare i dati inseriti durante i form sia della parte pubblica che della parte privata sia dell'utente che dell'amministratore;
2. per aggiornare costantemente il tipo di scelte che un utente può fare in base alle scelte effettuate dagli altri utenti.

Avendo suddiviso il progetto in cartelle, per ogni pagina che necessitasse di una validazione di un form abbiamo creato il suo corrispondente file javascript (nomeValidation.js) contenente i controlli utili per la validazione; per ogni pagina che invece necessitasse un aggiornamento continuo dei dati (vedi i corsi e gli ingressi nell'area utente), è stato creato il suo corrispondente file javascript (nome.js) contenente la chiamata AJAX al server.

4.1.5 XAMPP

Per testare la parte dinamica del sito, in particolare PHP e SQL, il gruppo ha deciso di utilizzare di XAMPP. Ciò ha reso possibile anche testare il sito con dispositivi mobili, aprendo la porta 80 del proprio pc e rendendo il tutto accessibile nella rete locale. Tramite questo il gruppo ha potuto testare il sito provando meglio i bottoni con le proprie dita, cosa che con lo strumento di ispezione di Chrome in modalità mobile era poco affidabile.

4.1.6 Ispezione di Chrome

La modalità ispezione di Chrome è buona per testare CSS e HTML, in quanto permette di visionare graficamente molti degli attributi dei fogli di stile, facilitando il debugging. Inoltre, permette di fare cambiamenti al CSS senza intaccare i file originali, consentendo di provare idee e metodologie di approccio di cui non si è particolarmente sicuri (come per esempio la scelta dei colori).

4.2 Funzionamento generale

4.3 Intrusività

Di seguito verrà descritto il livello di intrusività dei linguaggi rispetto al codice sorgente HTML:

- **CSS:** non intrusivo. Tutto il codice CSS è stato inserito in un file a parte, evitando del tutto pratiche come il CSS inline o embedded.
- **PHP:** non intrusivo. Sebbene ci sia del codice HTML all'interno di alcuni file PHP, la corrispondente pagina HTML non contiene codice PHP. La motivazione per la quale c'è del codice HTML in file PHP è che spesso i dati, essendo molto dinamici, cambiano spesso in base sia all'input dell'utente che dell'amministratore. Dunque in base alla presenza o meno di alcuni di questi (per esempio le date d'ingresso) possono comparire o meno determinati elementi HTML.
- **Javascript:** non intrusivo. Tutto il codice javascript è stato inserito in un file a parte.

5 Validazione

La validazione delle pagine è fondamentale per garantire un sito web accessibile e aderente agli standard. In particolare sono stati utilizzati i seguenti strumenti per la validazione automatica del sito.

5.1 Strumenti usati

Per la validazione dei file HTML sono stati utilizzati due servizi: TotalValidator⁴ e il validatore di W3C⁵. Per il CSS del sito è stato usato il validatore CSS⁶ sempre offerto dal W3C. Infine, per il PHP ed il Javascript, anche se non richiesto, abbiamo utilizzato rispettivamente PhpCodeChecker⁷ ed Esprima⁸.

5.1.1 TotalValidator

Questo tool ha permesso di validare tutti i file .html . Siamo consapevoli del fatto che compaiano degli errori poichè molte pagine vengono utilizzate dal PHP per produrre codice sorgente valido.

5.1.2 W3C HTML Validator

Lo strumento di validazione di W3C ci ha consentito di validare anche l'HTML prodotto dalle pagine PHP, poichè permette di incollare direttamente il codice sorgente. È il motivo per cui è stato utilizzato per validare tutto il codice HTML.

5.1.3 W3C CSS Validator

Abbiamo utilizzato anche il servizio di validazione del CSS del W3C così da assicurarci il rispetto il più rigorosamente possibile dello standard.

5.1.4 PhpCodeChecker

Attraverso l'utilizzo di questo tool è stata verificata la sintassi di tutti i file PHP. È stato molto utile in quanto essendo il sito molto dinamico buona parte del sito web è scritta con questo linguaggio.

5.1.5 Esprima

Tramite l'utilizzo di questo strumento è stata controllata la sintassi di tutti gli script javascript utilizzati, come la validazione dei campi lato client o le chiamate HTTP AJAX.

⁴<https://www.totalvalidator.com/>

⁵<https://validator.w3.org/>

⁶<http://www.css-validator.org/>

⁷<https://phpcodechecker.com/>

⁸<https://esprima.org/demo/validate.html>

6 Organizzazione del lavoro

Il lavoro del progetto *Adrenaline Motocross Park* è stato suddiviso nel seguente modo:

- Brugnolaro Filippo:
 - Parte dei file HTML della parte pubblica
 - Maggior parte dei file Javascript
 - Validazione HTML e Javascript
 - Prova con screen reader
 - Stesura della relazione
- Cavaliere Alessandro:
 - File HTML della parte privata dell'amministratore
 - Maggior parte dei file PHP della parte privata dell'amministratore
 - Parte dei file Javascript
 - Parte del database SQL
 - Parte dei file CSS
- Gambirasio Leonardo:
 - Parte dei file HTML della parte pubblica
 - Parte dei file PHP
 - Parte dei file CSS
 - Validazione CSS
 - Test dei browser
- Simionato Riccardo:
 - File HTML della parte priva dell'utente
 - Maggior parte dei file PHP della parte privata dell'utente
 - Parte del database SQL
 - Parte dei file CSS
 - Validazione dei file PHP

A Note per la correzione

In questa sezione vengono illustrate alcune informazioni utili per i valutatori.

A.1 Credenziali

Vengono fornite le credenziali dell'account amministratore:

- Username: admin
- Password: admin

Vengono fornite le credenziali dell'account utente:

- Username: utente
- Password: utente

É in ogni caso sempre possibile creare nuovi utenti tramite l'apposito form di registrazione.

A.2 Popolamento

Per avere una reale presentazione del sito, sono già stati inseriti alcune date, alcuni tracciati e alcune moto noleggiabili.

I dati inseriti non devono per forza corrispondere con la realtà e per tanto non devono essere considerati come difetti del sito.