



Museo Ferrari

Museo Ferrari

Corso di Tecnologie Web

A.A. 2018-2019

Relazione di progetto

Componenti	Marco Chilese Samuele Gardin Matteo Marchiori Giovanni Peron
Referente	Giovanni Peron

Link al sito:

[http...](http://)

Indice

1 Abstract	1
2 Utenti Destinatari	2
3 Sviluppo	3
3.1 Progettazione	3
3.2 Design	3
3.3 HTML e CSS	4
3.4 Apache	5
3.5 MySQL e PHP	5
3.6 JavaScript	6
4 Accessibilità	8
4.1 Colori	8
4.2 Immagini	8
4.3 Scelte progettuali	8
4.4 Navigazione nel sito	9
4.5 Mobile	10
4.6 Test accessibilità	12
4.6.1 Test daltonismo	12
4.6.2 Contrasto colori	14
4.6.3 Test Total Validator Tester	15
5 Installazione	16
5.1 Requisiti	16
5.2 Configurazione	16
6 Organizzazione interna	17
6.1 Gruppo	17
6.2 Ambiente di sviluppo utilizzato	17

1 Abstract

Il progetto sviluppato si propone di realizzare un sito Web che illustri, principalmente, il catalogo, le mostre correnti e future, e tutte le informazioni di un museo dell’automobile che espone modelli di auto Ferrari.

Il sito è a scopo informativo, permette all’utente di conoscere semplicemente quali auto siano esposte nel museo, quali mostre a tema particolare siano in corso e quali lo saranno prossimamente. Nel sito sono inoltre disponibili le informazioni circa orari, luogo e biglietti. È inoltre possibile, mediante un form, eseguire la richiesta di prenotazione di uno o più biglietti, generando un documento PDF da portare con sé il giorno della visita per l’effettivo acquisto.

L’intero sito è stato sviluppato tenendo a mente la separazione tra struttura, presentazione e comportamento, e le regole di accessibilità.

2 Utenti Destinatari

Il sito Web è rivolto ad un ampio pubblico di utenti, appassionati del settore e non. I contenuti sono esposti in modo chiaro e informale, facilitando quindi l'accesso ai contenuti anche da parte dei meno esperti.

Per interagire con il sito web del museo non è richiesta la registrazione. Se invece si desidera formulare una richiesta di prenotazione dei biglietti di visita, è necessario inviare i propri dati mediante un form. È inoltre possibile, se necessario, contattare lo staff in caso di necessità.

3 Sviluppo

3.1 Progettazione

Nella fase iniziale, grazie ai confronti avvenuti in alcune riunioni, il gruppo ha definito le linee guida da seguire nello svolgimento del progetto. Sono state scelte le tecnologie da utilizzare e sono stati divisi i compiti, nel modo più equo possibile, tra i membri del gruppo. È stato scelto di non utilizzare framework, ma di implementare la presentazione esclusivamente tramite CSS puro. Per facilitare la realizzazione di pagine responsive, è stato scelto di utilizzare un layout grafico righe/colonne, implementato in CSS.

Nella prima fase lo sviluppo del sito si è concentrato sul creare la struttura delle pagine in HTML e nell'aggiungere ad esse le regole di stile, creando prima il file *style.css* e poi *style_desktop.css*, secondo la logica mobile first; subito dopo è stato aggiunto anche lo stile per la stampa. In seconda fase è stata aggiunta tutta la struttura dinamica del sito ed è stato quindi creato anche il database. Infine, in una terza fase, sono state implementate tutte le accortezze per garantire l'accessibilità del sito e sono state svolte le attività di verifica della qualità del sito. La parte Javascript del sito è stata realizzata nella prima e nella terza fase di sviluppo del sito, e si è cercato di usare il più possibile funzioni e costrutti di linguaggio compatibili con lo standard ECMAScript 3, in modo da risultare compatibili con la più vasta gamma di dispositivi possibile.

Riteniamo opportuno precisare che è stato scelto di evitare l'implementazione della parte di gestione dei contenuti del sito, caricamento di articoli immagini ecc. Questa parte sarebbe stata essenziale nella realtà, se il sito fosse stato realizzato per essere venduto ad un qualsiasi cliente, ma essendo un progetto puramente a scopo accademico, abbiamo evitato di realizzare tale parte del sito sostanzialmente per non aggiungere inutile complessità ad esso e soprattutto per motivi di tempo.

3.2 Design

Per quanto riguarda i principi di design, il gruppo ha cercato di mantenere l'aspetto il più possibile semplice, pulito e ordinato. Il sito presenta un layout per desktop ed un altro per i dispositivi mobile. Il primo è stato pensato per mantenere un aspetto standard per tutte le tipologie di schermo, infatti è stato scelto di limitare la larghezza della sezione principale in modo da rendere piacevole la lettura anche agli utenti con schermi dal rapporto larghezza/altezza più alti. Il layout mobile, invece, è stato progettato per raggiungere una larghezza minima di 264px, a queste

condizioni restrittive il menù principale viene nascosto e diventa utilizzabile attraverso un "*burger*" button: il contenuto viene disposto in un'unica colonna prestando particolare attenzione a mantenere la giusta dimensione del testo.

Il contenuto di tutte le pagine è stato strutturato in modo da fornire solo le informazioni strettamente necessarie allo scopo della pagina stessa, cercando di disporle nella maniera più comprensibile possibile seguendo costantemente un ordinamento basato sull'importanza dell'informazione rispetto all'argomento relativo alla pagina. È stato scelto un menù di navigazione ad un singolo livello cercando anche di minimizzare l'ampiezza per non causare disorientamento agli utenti, rispettando la cosiddetta regola "*dal particolare al generale*".

Sia per il layout desktop, che per il layout mobile, sono stati selezionati pochi colori prestando attenzione a disporli con lo scopo di creare un adeguato contrasto in modo da facilitare la lettura a tutti gli utenti. Durante tutta la fase di progettazione e realizzazione del sito uno degli obiettivi più importanti per il gruppo è stato quello di raggiungere il più alto livello di accessibilità possibile.

3.3 HTML e CSS

La struttura del sito è stata creata adottando il linguaggio XHTML per garantire una compatibilità stabile e consolidata con tutti i browser e minimizzare il rischio di possibili problemi di accessibilità dati da HTML5.

I file presenti nel progetto, sebbene salvati con estensione .html, non sono completi, ma rappresentano dei pezzi di struttura della pagina. Dentro a tali pezzi sono presenti delle keyword identificate da due asterischi (*) posti all'inizio e alla fine di ciascuna keyword. Le keyword vengono processate lato server prima che la pagina venga restituita al client, e hanno diversi compiti, tra cui quello molto importante di evitare il più possibile sovrastruttura. Se un pezzo di struttura corrispondente a una parola chiave non ci deve essere, essa viene semplicemente cancellata. La completa separazione tra struttura e presentazione è stata raggiunta utilizzando tre diversi fogli di stile: uno per i dispositivi mobili, uno per il desktop ed uno per la stampa delle pagine del sito.

Abbiamo adottato un strategia mobile first realizzando quindi prima l'aspetto mobile e integrando in seguito il layout desktop. Per questo motivo il primo foglio di stile relativo al layout mobile, *style.css*, è quello il più ricco, mentre *style_desktop.css* contiene solo regole per adattare il sito ad un layout desktop. In entrambi i layout abbiamo utilizzato uno schema a colonne per disporre il contenuto nelle pagine. Si è cercato dunque di minimizzare il numero di colonne per non complicare l'aspetto generale del sito che vuole essere il più lineare e semplice. Per ottenere una disposi-

zione migliore di alcuni elementi, il sistema prevede che ogni contenitore occupante un certo numero di colonne potesse ospitare a sua volta altri elementi occupanti un certo numero di colonne, in modo ricorsivo. Tutto il codice HTML e CSS che è stato prodotto è stato validato tramite gli strumenti forniti da W3C¹². Il codice HTML è conforme allo standard XHTML 1.0 Strict. Il codice CSS risulta valido rispetto alla specifica CSS3 + SVG.

3.4 Apache

A livello di server http sono state usate due direttive poste in un file .htaccess, ovvero la DirectoryIndex e la FallbackResource.

La prima consente di impostare un file da cui far partire l'elaborazione nel caso la richiesta effettuata sia l'homepage del sito Web. La seconda viene normalmente usata nel caso di richieste a risorse inesistenti. Nel nostro caso la direttiva è stata usata in modo molto ampio: ogni risorsa richiesta che non è l'homepage ricade in questa seconda istruzione. Viene usata dal gruppo in questo modo per implementare il meccanismo tramite controller centrale delle richieste realizzato in un unico file PHP.

3.5 MySQL e PHP

Il lato server è una parte molto importante e consiste nell'elaborazione dei dati che saranno poi inviati al client. Il database che viene utilizzato è stato sviluppato per la gestione delle seguenti informazioni:

- auto esposte;
- biglietti;
- eventi;
- utenti.

Come DBMS abbiamo deciso di utilizzare MySQL e come engine di storage InnoDB che ci garantisce un buon compromesso tra dimensioni massime (64TB) e la sicurezza nel mantenere la consistenza dei dati.

Non è stato necessario costruire alcun trigger nel DBMS, non ne è stata vista la necessità data la semplicità del sito implementato. Alcuni controlli di coerenza sono

¹<https://validator.w3.org/>

²<http://jigsaw.w3.org/css-validator/>

stati fatti direttamente lato server in PHP, di modo da assicurarsi che un utente già registrato per la prenotazione di biglietti potesse iscriversi successivamente alla newsletter del sito, oppure potesse comprare biglietti dopo esserli iscritto alla newsletter anche senza registrare completamente i dati anagrafici nella base di dati. Per quanto riguarda PHP il sito è stato progettato per essere molto dinamico nella costruzione del contenuto. Come da progettazione è stato adottato un modello con classi di utilità che forniscono molti metodi statici, ovvero classi per interagire con il database, inviare email e generare qrcode, mentre la gestione delle singole pagine avviene attraverso dei controller php che coordinano il modo in cui le richieste vengono elaborate. Vi è inoltre un unico file di partenza che funge da smistatore di richieste. Tutte le richieste vengono inoltrate da Apache verso questo file che si occupa di chiamare il controller corretto in base all'uri della risorsa richiesta. PHP elabora i pezzi di struttura html sostituendo le keyword menzionate sopra, in modo da avere dati aggiornati, dinamici ed evitando in parte il problema della sovrastruttura. Viene usato tale sistema per la segnalazione degli errori nella compilazione dei form all'utente e per mantenere i dati inseriti dopo l'invio della richiesta attraverso form html.

3.6 JavaScript

Particolare attenzione è stata data a JavaScript, esso è linguaggio di scripting di tipo client side che permette una maggiore accessibilità e dinamismo. È stato utilizzato per compiti rilevanti ma non fondamentali. Tale scelta trova giustificazione nel fatto che il sito deve restare usufruibile anche nel caso in cui venga consultato da dispositivi che non supportano JavaScript, o che hanno disabilitato gli script lato client.

Pertanto i compiti che gli sono affidati sono i seguenti:

- gestione dinamica della navbar a "burger";
- gestione dinamica della posizione dei bottoni *avanti* e *indietro* della pagina "*Modelli Esposti*".
- controllo dei dati in input nei form della pagina "*Biglietti*";
- controllo dei dati in input nei form della pagina "*Info e Contatti*".

Per il primo punto qualora javascript fosse disabilitato il menù sarebbe comunque usufruibile, rimarrebbe semplicemente sempre aperto senza fornire il pulsante burger di apertura. Per gli ultimi due punti, il controllo dei dati non è unico: viene

effettuato anche lato server attraverso controlli in PHP, per garantire la correttezza dei dati qualora JavaScript fosse disabilitato. Per il secondo punto non è stata posta un'alternativa a javascript, essendo la presenza dello stile dipendente dalla presenza di tutti i bottoni o meno, ma essendo un problema puramente estetico non si è vista la necessità di porre un'alternativa (la navigazione non viene compromessa e sono presenti ugualmente tutti gli strumenti disponibili). Nel controllo dei dati in input sui biglietti è stato sviluppata una verifica in AJAX della data immessa per la prenotazione della mostra, questo perché non sarebbe possibile sapere a priori quale mostra viene scelta dall'utente. Non si tratta propriamente di AJAX poiché viene eseguito in modo sincrono, per impedire l'invio della richiesta di prenotazione dei biglietti al server prima della verifica. Il controllo viene svolto in modo equivalente lato server, anche se in questo caso può risultare più fastidioso perché l'utente non riesce a sapere a priori in quali date sarà disponibile una mostra. Un'alternativa sarebbe stata quella di predisporre un elenco con le date disponibili relative a ogni mostra, cosa che abbiamo deciso di non fare per un motivo di accessibilità e scalabilità del sito.

4 Accessibilità

4.1 Colori

La gamma cromatica utilizzata nel sito web sviluppato è stata ristretta in modo tale da garantire un buon contrasto tra contesto e testo. Per arrivare alle nostre conclusioni per quanto riguarda i colori ci siamo serviti di un'estensione per Google Chrome, *Colorblindly*. Ci ha permesso di testare il nostro sito con i quattro tipi di daltonismo più diffusi: protanopia, deuteranopia, tritanopia e acromatopsia. In questo modo, il sito risulta accessibile a tutte le categorie di utenti con difficoltà visive.

Dove possibile si adottano colori definiti *web safe*³, in modo tale da garantire una corretta visualizzazione del sito su un maggior numero di dispositivi.

Per verificare l'adeguatezza del contrasto dei colori usati per il testo ci siamo serviti dello strumento *Colour Contrast Analyser*. Con quest'ultimo è stato possibile verificare che ogni abbinamento tra colore di sfondo e colore del testo rispettasse un rapporto di contrasto tale da essere conforme ai seguenti criteri:

- 1.4.3 Contrast (Minimum)
- 1.4.6 Contrast (Enhanced)
- 1.4.11 Non-text Contrast

presenti nelle linee guida WCAG 2.1.⁴

4.2 Immagini

Le immagini all'interno del sito sono tutte accompagnate dall'attributo *alt* al quale è stata assegnata una descrizione esaustiva ma non eccessivamente dettagliata dell'immagine. Ciò rende le immagini accessibili anche alle categorie di utenti con difetti visivi, poiché lo screen-reader si avvarrà di tali descrizioni.

4.3 Scelte progettuali

In corso d'opera abbiamo fatto delle modifiche per raggiungere il più alto livello di accessibilità possibile. Per rappresentare i dati tecnici delle auto esposte subito avevamo optato per la rappresentazione in forma tabellare, ma ci siamo presto resi conto che poteva essere un ostacolo per uno screen-reader, quindi abbiamo scelto la

³<https://www.color-hex.com/216-web-safe-colors/>

⁴<https://www.w3.org/TR/WCAG21/>

rappresentazione attraverso elenco.

La navbar nativamente era di colore giallo come il logo, ma dopo alcuni test ci siamo resi conto che non era la scelta ideale per un utente con problemi di acromatopsia, quindi abbiamo usato il grigio e il nero.

Avevamo dato per scontato che mettendo la mappa del luogo del museo fosse chiaro dove recarsi, ma non avevamo pensato che uno screen-reader avrebbe potuto aver problemi nell'interpretarla correttamente, è stato quindi aggiunto all'interno della pagina in questione l'indicazione testuale dell'indirizzo del museo.

4.4 Navigazione nel sito

La navigazione nel sito è garantita, in primo luogo, da una semplice *navbar* che non include sottomenù, come illustrato in §3.2. Sono direttamente visibili e accessibili tutte le pagine del sito web:

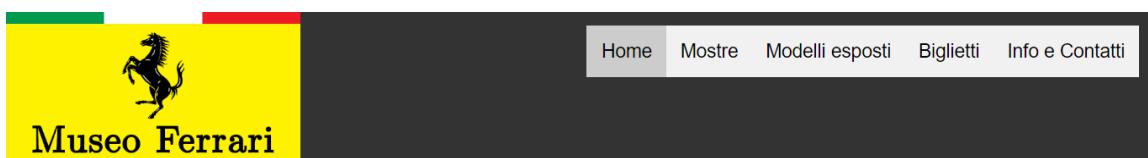


Figura 1: Header del sito, in particolare visto dalla pagina mostre.

Inoltre, il posizionamento corrente all'interno del sito web è chiarito grazie all'evidenziazione della pagina corrente all'interno della navbar:

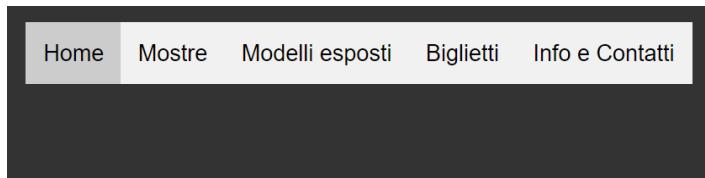


Figura 2: Dettaglio: la pagina corrente è evidenziata rispetto alle altre.

Oltre a ciò, sotto l'intestazione del sito è presente l'indicazione testuale sulla corrente posizione all'interno del sito ("breadcrumbs"):

Ti trovi in: Mostre

Figura 3: Dettaglio: *breadcrumbs*.

Per quanto concerne i link presenti nel sito, è stato fatto in modo che link "visitati" e

"ancora da visitare" siano facilmente distinguibili. Un esempio è visibile di seguito.



Figura 4: Dettaglio: link da visitare e visitati. Il link centrale è stato già visitato mentre il primo e l'ultimo risultano ancora da esplorare.

Per rendere efficiente la navigazione all'interno del sito anche a utenti con difficoltà visive, è stato fatto uso di *tabindex* per consentire un'agile navigazione all'interno del sito. Ogni immagine è inoltre accompagnata da un *alt* che descrive in modo soddisfacente l'immagine in oggetto.

È stato deciso di non introdurre *accesskey* poiché gli shortcut stabiliti avrebbero potuto entrare in conflitto con i comandi definiti di default dai browser.

È stato inoltre inserito un bottone che consente di tornare facilmente in testa alla pagina.

4.5 Mobile

Per cercare di rendere il sito compatibile con la maggior parte dei dispositivi mobile (smartphone, tablet) abbiamo deciso di implementare l'accessibilità con CSS puro.

Il sito in modalità mobile si presenta così:



Figura 5: Pagina Home in versione mobile.



Figura 6: Pagina dei modelli esposti in versione mobile.



Figura 7: Pagina dei modelli esposti in versione mobile.

4.6 Test accessibilità

Con lo scopo di verificare e accertarsi che i livelli di accessibilità raggiunti fossero soddisfacenti sono stati effettuati dei test. Tali test non hanno il presupposto di essere esaustivi, bensì informativi.

Nei prossimi paragrafi mostriamo, con l'ausilio di alcuni screenshot, i risultati ottenuti.

4.6.1 Test daltonismo

Riportiamo di seguito i test effettuati sulla home page del sito per quanto riguardano i difetti visivi relativi alla cecità completa o parziale ai colori, noti come daltonismo.

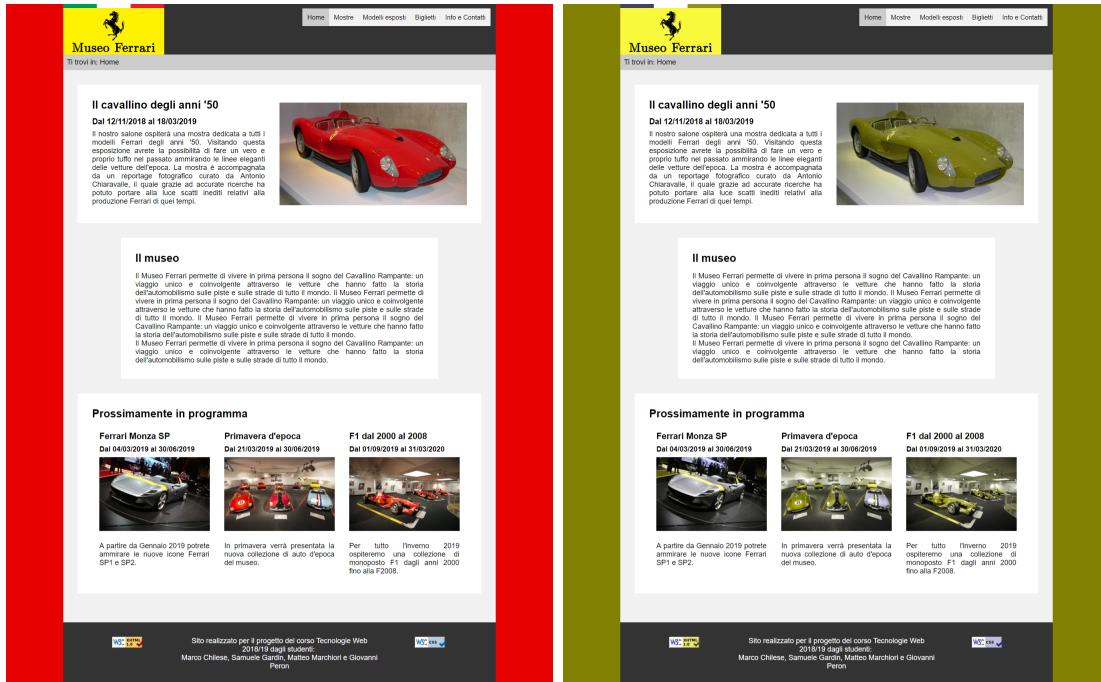


Figura 8: Test daltonismo protanopia

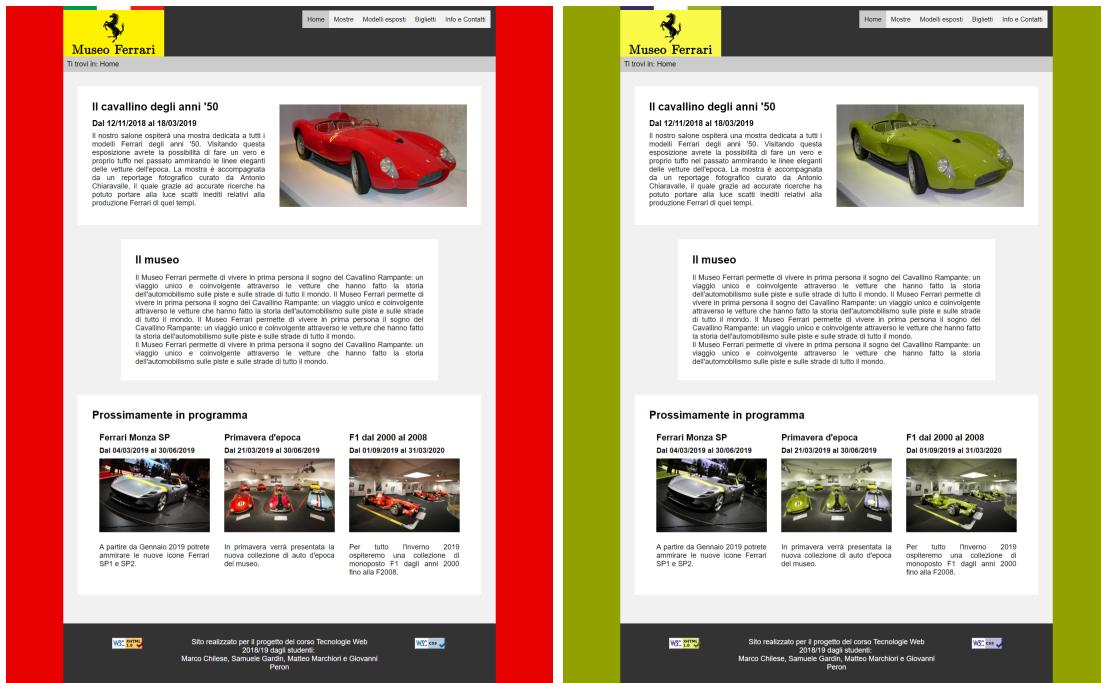


Figura 9: Test daltonismo deuteranopia

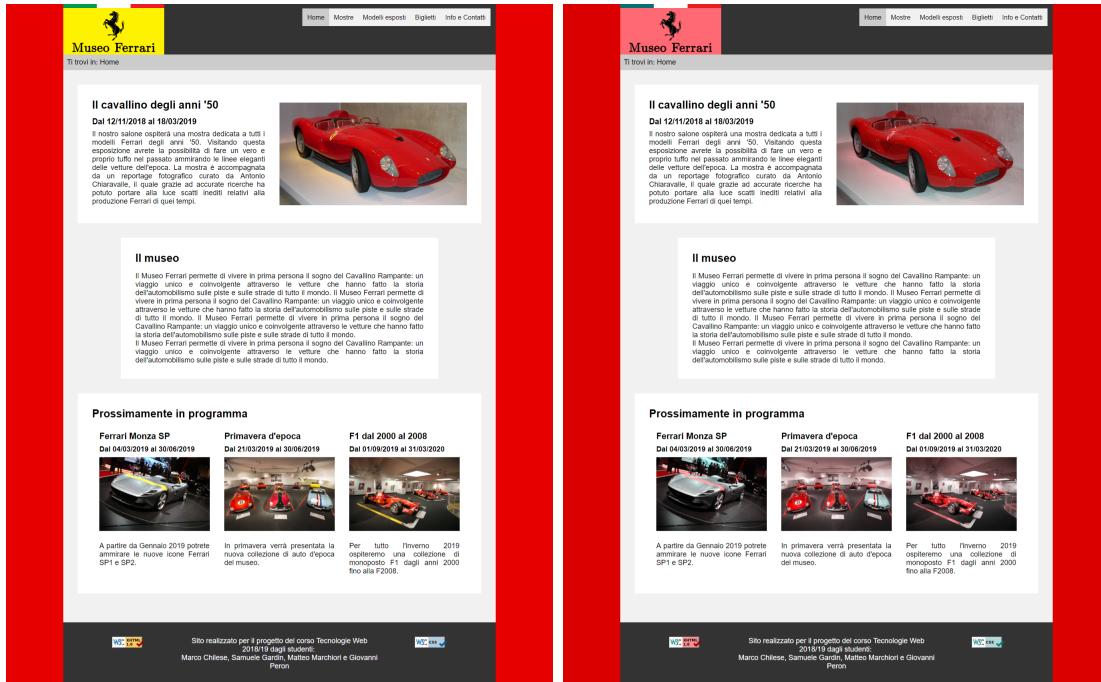


Figura 10: Test daltonismo tritanopia

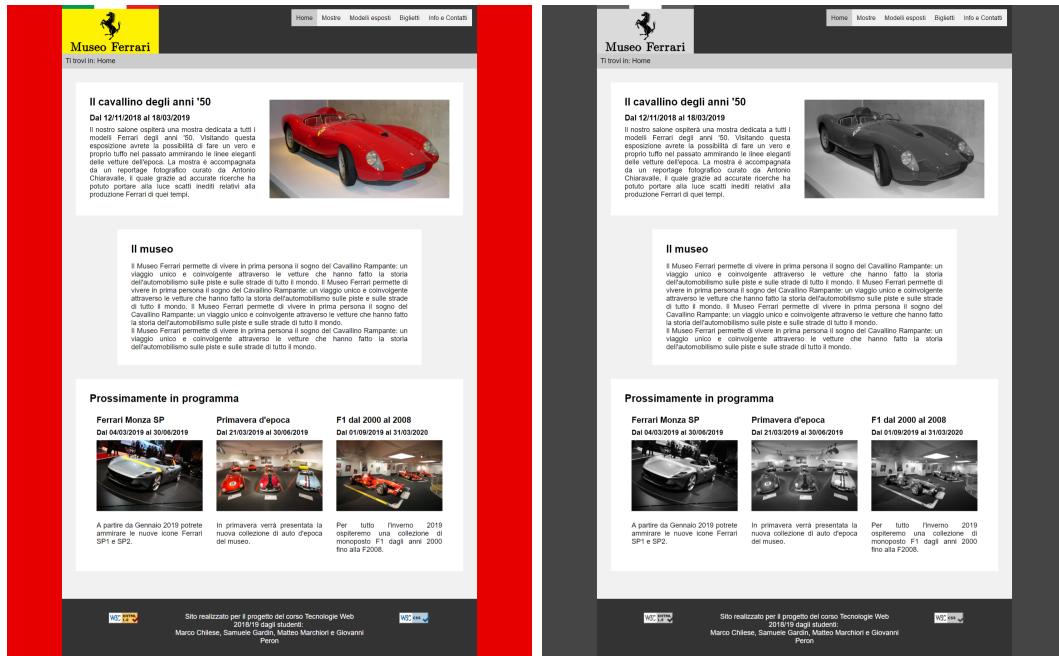


Figura 11: Test daltonismo acromatopsia

4.6.2 Contrasto colori

La seguente tabella espone i risultati ottenuti con lo strumento *Colour Contrast Analyser* relativi a ogni abbinamento tra colore di sfondo e colore del testo presente

nel sito. Tutte le coppie di colori scelti rispettano i criteri 1.4.3 Contrast (Minimum), 1.4.6 Contrast (Enhanced) e 1.4.11 Non-text Contrast, presenti nelle linee guida WCAG 2.1.

Colore di sfondo	Colore del testo	Contrast ratio	Utilizzo nel sito
#333333	#FFFFFF	12.6 : 1	Simbolo hamburger del menù di navigazione, footer, label o legend, pulsanti
#CCCCCC	#000000	13.1 : 1	Pagina corrente nel menù di navigazione, form delle pagine biglietti e info e contatti
#F1F1F1	#000000	18.6 : 1	Pagine non visitate nel menù di navigazione
#F1F1F1	#800080 (purple)	8.3 : 1	Pagine visitate nel menù di navigazione
#FFFFFF	#000000	21 : 1	Testo delle pagine del sito
#FFFFFF	#0000FF (blue)	8.6 : 1	Link non visitati nel sito ad esempio nella pagina mostre
#FFFFFF	#800080 (purple)	9.4 : 1	Link visitati nel sito ad esempio nella pagina mostre
#CCCCCC	#990000	5.6 : 1	Messaggi d'errore nei form
#CCCCCC	#006600	4.5 : 1	Messaggio stato di invio nel form biglietti

4.6.3 Test Total Validator Tester

Grazie all'applicazione *Total Validator Tester*⁵ e al sito *Achecker*⁶, è stato verificato il livello di accessibilità, per ogni pagina del sito. Dai test risulta che il sito è conforme alle linee guida dettate dallo standard WCAG 2.0 Level AA.

⁵<https://www.totalvalidator.com/downloads/index.html>

⁶<https://achecker.ca/checker/index.php>

5 Installazione

In questa sezione vengono specificati i requisiti necessari per la corretta installazione del sito.

5.1 Requisiti

- Server HTTP Apache;
- PHP versione 7.0 o superiore;
- Un database engine compatibile, ad esempio MariaDB.

5.2 Configurazione

- Supporto alle istruzioni relative a DirectoryIndex e FallbackResource di Apache;
- Estensione mysqli di PHP.

Non è invece richiesta l'abilitazione del modulo rewrite di Apache.

È possibile usare un server HTTP diverso da Apache, ma in tal caso bisogna fare in modo che tutte le richieste puntino al file header.php, in modo da preservare il comportamento del sito. In caso il sito venga spostato in una directory diversa, bisogna assicurarsi che le istruzioni DirectoryIndex e FallbackResource puntino al file header.php.

6 Organizzazione interna

Nella seguente sezione si analizza l'organizzazione interna del gruppo, la relativa suddivisione dei compiti e le caratteristiche dell'ambiente utilizzato per lo sviluppo del progetto.

6.1 Gruppo

I compiti all'interno del gruppo sono stati divisi nel seguente modo:

- Referente: Giovanni Peron;
- Struttura HTML: Marco Chilese, Matteo Marchiori, Samuele Gardin, Giovanni Peron;
- Presentazione CSS: Marco Chilese, Matteo Marchiori, Samuele Gardin, Giovanni Peron;
- Progettazione database: Samuele Gardin, Marco Chilese;
- PHP:
 - definizione: Matteo Marchiori;
 - implementazione: Marco Chilese, Matteo Marchiori, Samuele Gardin, Giovanni Peron;
- JavaScript: Matteo Marchiori;
- Accessibilità: Marco Chilese;
- Contenuti: Samuele Gardin, Giovanni Peron;
- Relazione: Marco Chilese, Giovanni Peron;
- Testing: Marco Chilese, Matteo Marchiori, Samuele Gardin, Giovanni Peron.

6.2 Ambiente di sviluppo utilizzato

- SO: Ubuntu 16.04 LTS⁷, MacOS 10.14.3, Windows 10 64bit;
- Editor di testo: Visual Studio Code, Notepad++;
- Server locale:

⁷Kernel 4.15.0-43-generic

- Apache version 2.4.18;
- MySQL version 14.14 Distrib 5.7.25;
- PHP version 7.2.14-1.