

CINEPado

Localhost:porta/tdifant/HTML/index.php

Progetto per il corso di Tecnologie Web, prof.ssa Ombretta Gaggi

Corso di Laurea in Informatica, Università degli Studi di Padova

Componenti del gruppo:

Luca Annicchiarico	luca.annicchiarico@studenti.unipd.it	matricola 1229143	
Tommaso Di Fant	tommaso.difant@studenti.unipd.it	matricola 1217932	(referente)
Giovanni Moretti	giovanni.moretti.2@studenti.unipd.it	matricola 1217655	
Alberto Nicoletti	alberto.nicoletti@studenti.unipd.it	matricola 1224821	

Credenziali utenti:

Utente semplice: login: **user**, password: **user**

Utente amministratore: login: **admin**, password: **admin**

Indice

Installazione	3
1. Analisi	3
Contesto	3
Utenti	3
Ricerche.....	3
2. Progettazione.....	4
Ampiezza e profondità.....	4
Struttura delle pagine	4
Struttura del sito	4
Funzionalità.....	6
Database SQL	6
3. Fase di realizzazione.....	7
Struttura.....	7
Presentazione	7
Stile generale	7
Schermi piccoli	8
Stampa	8
Comportamento Javascript.....	8
Comportamento Php	9
Pagine php	10
4. Accessibilità.....	11
Link	11
Colori.....	12
Immagini	12
Test.....	12
Validazione.....	12
Screen Reader	13
Focus	13
Browser	13
Dispositivi.....	14
Strumenti Accessibilità.....	14
Errori	14
6. Organizzazione del lavoro	16

Installazione

Per poter effettuare connessioni al Database la classe DBAccess fa uso di un file di configurazione `/php/config.json`. Esso contiene le credenziali di accesso al DB relative al proprio utente; assicurarsi che nome utente e path al file contenente la password siano corrette.

Prima di utilizzare le funzionalità che fanno uso del database eseguire lo script di creazione e popolamento in `/SQL/festival.sql`.

Per il corretto funzionamento del codice di errore HTTP/S 404 definire la path assoluta al file contenente la pagina di risposta definita in `/HTML/.htaccess` come `/nome_utente/HTML/404.php`.

1. Analisi

Contesto

CinePado è un piccolo festival di cinema che si svolge a Padova in una settimana a settembre. L'obiettivo del festival è promuovere il cinema come arte, ciò si riflette nella scelta delle proiezioni e nel promuovere registi "in erba" attraverso il CPContest. Lo sviluppo di un sito web di CinePado nasce da alcune esigenze:

- Dare la possibilità di acquistare i biglietti on-line.
- Permettere una facile consultazione dei film proposti e del programma delle proiezioni.
- Creare un sistema per la gestione delle candidature al CPContest.
- Farsi conoscere ed aumentare la propria utenza.

Utenti

Data la tipologia di film proposti alle proiezioni, si prevede che l'utenza sia composta da persone di ogni età, ma soprattutto la fascia 20-30, appassionati del cinema più come arte che intrattenimento. Tale pubblico si è fidelizzato, negli ormai 5 anni di festival. Il sito si propone quindi di facilitare il reperimento di informazioni per utenti che già sanno cosa vogliono o ne hanno almeno un'idea (pesca a tiro perfetto/trappola per aragoste). Il sito vuole comunque raggiungere i possibili futuri utenti del festival. In particolare, la home funge da vetrina, mostrando film popolari, le date in cui si svolge il festival e alcune foto.

Da un paio di anni il traffico web è effettuato in maggior parte da dispositivi mobili. Ciò vale anche per questo sito, che si prevede essere raggiunto da utenti che da smartphone si stanno informando sulle proiezioni. Il probabile alto utilizzo del sito da smartphone è dato anche dal fatto che il target previsto di persone è maggiormente nella fascia 20-30 anni. Per le pagine riguardanti il contest si prevede un maggiore utilizzo da pc, essendo il suo contenuto da leggere più attentamente quindi con calma e usando un pc. Anche le pagine per gli utenti admin saranno più utilizzate da pc per lo stesso motivo.

Ricerche

Si prevede che il sito debba rispondere ad alcune ricerche sui motori di ricerca (i meta tag keywords sono scritti in previsione di ciò). Alcuni esempi di possibili ricerche:

Festival di cinema Padova, cinepado, cinema Padova, festival di cinema

2. Progettazione

Ampiezza e profondità

Il sito presenta un menù di 6 voci, un numero sufficiente per distinguere i vari tipi di informazioni che si possono trovare, ma non troppo elevato (che rischierebbe di confondere o mettere a disagio l'utente).

La profondità raggiunge i 2 livelli, nelle pagine di lista dei film e lista delle proiezioni. Il massimo numero di click (al netto di indecisioni, torna indietro, ecc..) si ottiene quando un utente arriva alla pagina di acquisto di un biglietto, passando per la lista dei film, con 4 click. È un numero adeguato, anche dovuto al fatto che l'utente deve prima scegliere il film che vuole vedere. Nel caso abbia già le idee chiare, può andare alla pagina delle proiezioni direttamente, raggiungendo l'acquisto del biglietto con soli 2 click.

Struttura delle pagine

Ogni pagina è strutturata in un modo simile e segue il design a tre pannelli, che risponde alle domande per evitare il disorientamento (Dove sono? Dove posso andare? Di cosa si tratta?). Ogni pagina ha un contenuto diverso ma tutte le pagine presentano delle sezioni in comune:

Header

L'header, come intestazione della pagina, presenta sempre le stesse caratteristiche: un logo, una *navbar* composta da pagine web ad accesso rapido e un pulsante per il profilo utente. Le pagine web accessibili tramite l'header sono: Home, About us, Film, Proiezioni, Come Trovarci, Il Contest. Il tasto Utente permette di inserire le credenziali di accesso o procedere alla registrazione di un nuovo account. Una volta effettuato l'accesso, questo pulsante Utente mostra il nome dell'utente e permetterà di accedere alla pagina personale o di fare il logout.

Breadcrumb

Il breadcrumb è presente in tutte le pagine del sito, prima del contenuto della pagina stessa. È composto da una stringa testuale che indica la posizione dell'utente nella gerarchia del sito

Footer

Il footer è composto da una lista degli sponsor, l'elenco dei componenti del gruppo e la dicitura del copyright. Nel caso di un utente admin loggato, compare in grande un link per accedere alla sezione di admin.

Struttura del sito

Di seguito sono elencate le pagine web e il loro contenuto:

Home: È la prima pagina che l'utente visualizza, contiene cinque sezioni:

- Descrizione: una breve descrizione del festival.
- Film più popolari: una lista di film ordinati per numero di like dati dagli utenti del sito.
- Acquisto biglietti: una lista delle date delle proiezioni con link alla pagina delle proiezioni con filtro attivo sulla data selezionata.
- Dicono di noi: alcune recensioni sul festival tratte da giornali.
- Foto: tre foto che mostrano: il gruppo che organizza il festival, la piazza dove si proiettano i film e la premiazione del vincitore del CPCcontest di qualche anno fa. Ogni immagine è

corredata da un titolo che rimanda, rispettivamente, alle pagine: About Us, Come trovarci, Il Contest.

About us: La pagina About us fornisce informazioni sul festival e la sua storia e sugli organizzatori. Una sezione è dedicata all'accessibilità. Si conclude con un elenco di modi per contattare gli organizzatori.

Lista film: Questa pagina contiene una lista dinamica con un riquadro per ogni film in proiezione. Ogni riquadro contiene il titolo del film, il regista e permette di raggiungere la pagina con i dettagli del film e la pagina delle proiezioni del film. Se il film fa parte del CPContest, compare la dicitura "film in gara" e un tasto per dare un like al film. Inoltre, la lista può essere filtrata per titolo del film e appartenenza CPContest o meno.

Pagina Film: La pagina con i dettagli del singolo film, in particolare un'immagine tratta da un frame del film e una descrizione. Anche da questa pagina si può accedere alla lista delle proiezioni del film in questione. La pagina è un template e viene riempita dinamicamente con i dati del film richiesto presenti nel database.

Proiezioni: Questa pagina, simile alla pagina Lista Film, contiene una lista dinamica con un riquadro per ogni proiezione. Ogni riquadro contiene il titolo del film, il regista, l'eventuale dicitura "film in gara", la data e l'ora della proiezione. Dal riquadro si può andare alla pagina del film, per conoscerne i dettagli, o andare alla pagina di acquisto biglietto per la proiezione in questione. La lista può essere filtrata per titolo del film, data della proiezione, e l'appartenenza al CPContest o meno.

Acquisto biglietti: La pagina di acquisto dei biglietti può essere acceduta solo tramite la lista delle proiezioni. Nel caso non si sia effettuato alcun accesso, essa presenta una sezione che permette il login o la registrazione dell'utente. Se si è invece loggati, viene mostrato un riepilogo della proiezione selezionata ed un pulsante per acquistare il biglietto. L'acquisto del biglietto rimanda ad una pagina di feedback positivo con il riepilogo dell'ordine.

Come trovarci: Contiene tutte le informazioni necessarie per raggiungere il luogo in cui si svolgono le proiezioni, e in varie modalità (a piedi, in auto, in bus). Inoltre presenta un elenco di locali consigliati in cui bere e/o mangiare.

Pagina Contest: Contiene il regolamento del festival ed una form per inviare le proprie candidature, visibile solamente se loggati.

Pagina Utente: La pagina Utente permette ad ogni utente che ha effettuato l'accesso di visionare e modificare le proprie informazioni personali, visionare i biglietti acquistati e le candidature effettuate, con la possibilità di ritirare una candidatura, solo se non è stata ancora approvata.

Pannello admin: La pagina del pannello di controllo admin mostra agli utenti abilitati delle statistiche sull'utilizzo del sito. Come tutte le altre pagine admin, sulla sinistra è presente un menù per accedere alle altre funzionalità dell'admin:

- Visualizzare le candidature al CPContest con possibilità di accettare e rifiutare.
- Aggiungere, modificare o eliminare le proiezioni di film.
- Aggiungere, modificare o eliminare i film.
- Visualizzare i dati di ogni utente, ed i relativi biglietti acquistati. L'admin può modificare, eliminare o aggiungere biglietti agli utenti. Può anche eliminare gli account.

Pagina 404: La pagina 404 comunica all'utente la non presenza di una pagina web all'interno del sito e lo invita, tramite dei link, a tornare alla pagina home oppure andare alla sezione contatti.

Funzionalità

Le classi di utenti che navigano il sito sono tre, ognuna con le proprie funzionalità:

- **utente non registrato:** Ha la possibilità di vedere tutti i contenuti informativi del sito, tra cui: elenco proiezioni con data e orario, lista film (in gara e non), regolamento contest.
- **utente registrato:** Oltre alle funzionalità dell'utente non registrato può:
 - acquistare biglietti relativi alle proiezioni.
 - visualizzare il profilo personale con elenco dei biglietti acquistati e delle candidature proposte, con possibilità di modificare i propri dati, eliminare l'account e ritirare le candidature "in sospeso".
 - proporre un film come candidato al contest.
 - mettere like ai film in concorso.
- **utente admin:** Oltre alle funzionalità dell'utente registrato può accedere alla sezione admin dove è possibile:
 - Visualizzare la dashboard con statistiche relative al DB.
 - Visualizzare la lista delle candidature, con possibilità di approvare o scartare (eliminare) le candidature in sospeso e sospendere le candidature approvate. I film approvati, verranno presentati nell'elenco dei film in gara.
 - Aggiungere o rimuovere le proiezioni; oppure modificarne orario, data e film in proiezione.
 - Aggiungere o rimuovere i film; oppure modificarne i dati relativi.
 - Visualizzare o eliminare gli account degli utenti registrati; aggiungere, modificare o eliminare i biglietti associati ad ogni utente.

Si noti come non esista una classe di utenza particolare per registi e produttori interessati al CPContest.

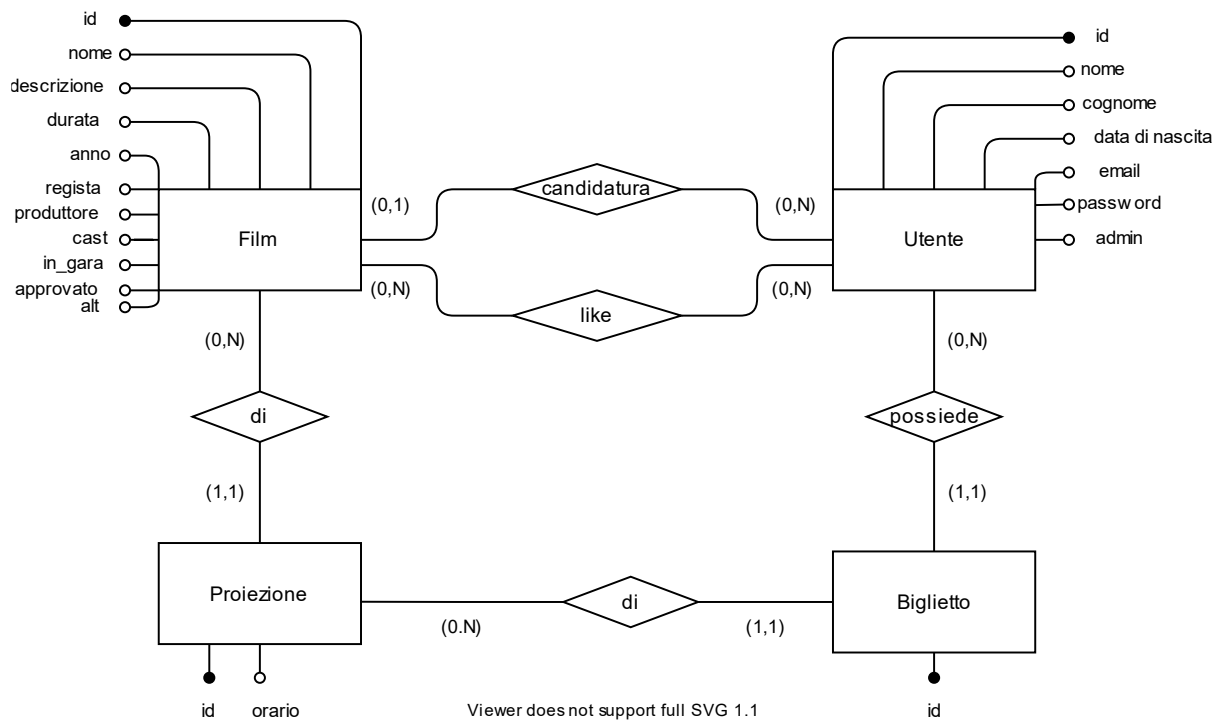
Database SQL

Il database è composto da quattro entità principali:

- **Film:** rappresenta un Film, o una Candidatura. Contiene una chiave esterna candidato, se definito identifica il Film come Candidatura e contiene l'Utente candidato. Oltre ai campi descrittivi stretti al concetto di film si possono distinguere:
 - **in_gara:** *bool*, definisce se il film è in gara o meno.
 - **approvato:** *bool*, distingue un film candidato in sospeso da uno approvato.
 - **alt:** *string*, definisce il contenuto dell'attributo alt relativo all'immagine relativa al film.
- **Utente:** rappresenta un *Utente*.
- **Proiezione:** rappresenta una *Proiezione*, chiave esterna a *Film*.
- **Biglietto:** rappresenta un *Biglietto*, mette in relazione *Utente* e *Proiezione* entrambe come chiavi esterne.

Oltre alle entità sopraelencate la relazione **Like** è stata trasformata in un'entità che mette in relazione *Utente* e *Film*.

Segue lo schema ER:



3. Fase di realizzazione

Le pagine sono scritte in HTML5, rispettando la sintassi XML. Nell'implementazione è stato seguito il principio di separazione tra struttura, presentazione e comportamento.

Struttura

Nell'implementare la struttura HTML del sito si è cercato di:

- Rispettare la correttezza semantica dei tag.
- Usare correttamente i meta tag.
- Mantenere il codice pulito e non usare tag superflui.
- Seguire regole nell'uso dei tag. Ad esempio: title dal particolare al generale, titoli (h1, h2) nel corretto ordine...
- Non usare tag obsoleti, deprecati o nocivi.
- Scrivere codice valido.

Presentazione

La presentazione è data da tre fogli di stile. "style.css" si applica a tutti gli schermi, se gli schermi hanno una larghezza inferiore ai 600px, agli stili precedenti si aggiungono quelli definiti in "small.css". Quando una pagina del sito viene stampata, si applicano gli stili definiti da zero in "print.css"

Stile generale

Scelte e obiettivi interessanti nella realizzazione del foglio di stile "style.css":

- Creare un sito visivamente pulito e ben organizzato.
- Usare un colore di bandiera e creare una palette che nel complesso crei un bel ambiente nel quale navigare.
- Creare un design fluido e che funzioni bene su vari browser e vari dispositivi.

- Le pagine Lista Film e Proiezioni hanno una struttura simile di lista che utilizza il layout flex. Questa lista è composta da una serie di piccole schede con le informazioni principali del film o della proiezione. Il layout flex permette una buona fluidità nella disposizione di queste schede.
- La pagina del singolo film usa un layout Grid per disporre le informazioni del film.

Altre scelte implementative degne di nota sono ben descritte nella sezione 4. Accessibilità.

Schermi piccoli

L'obiettivo del foglio di stile "small.css" è rendere più usabile e chiaro il layout su schermi (o finestre) di larghezza inferiore ai 600px. Per fare questo alcune regole di stile sono state parzialmente sovrascritte per essere meglio supportate, mentre altre sono state sovrascritte completamente per ottenere un layout che si conformi meglio alle dimensioni ridotte.

Alcune scelte che riflettono questo obiettivo sono:

- Al fine di rendere più usabile il menù, questo si sviluppa su tre righe diverse, con ogni link che prende il 50% di una riga; in questo modo non solo è più chiaro da vedere, ma i link sono anche più facilmente cliccabili.
- Il padding del content è stato ridotto per sfruttare al meglio il minor spazio a disposizione.
- Il layout grid della pagina "templateFilm.html" è stato riorganizzato in modo da svilupparsi verticalmente.
- La finestra di login/signup è stata ridimensionata in modo da prendere tutto lo schermo.
- Le form sono state rese più larghe per favorire la leggibilità e la facilità di compilazione.

Stampa

Per la stampa si è partiti dall'obiettivo di fare risparmiare inchiostro e carta (basandosi sul formato A4 sono state ristrette le pagine che "strabordavano" per solo un paio di righe). Altri accorgimenti:

- Rimozione delle immagini e dei colori, utilizzo di font con grazie.
- I link sono seguiti dall'url esteso della loro destinazione.
- Rimozione del menu nell'header.
- Il footer è stato mantenuto, non potendo eliminare gli sponsor, ed è giusto mantenere anche i nomi dei componenti del gruppo.
- Le immagini nella pagina del singolo film, essendo parte importante, sono state mantenute ma ridotte di dimensione.
- Tramite la proprietà *break-inside:avoid*; Si è cercato di evitare che venissero spezzate in due pagine parti di contenuto strettamente legate. Ad esempio il titolo di un film ed il suo regista nella Lista Film.

Comportamento Javascript

Alcuni contenuti dinamici vengono gestiti da javascript, per esempio la freccia "Torna su" compare quando l'utente ha scrollato più di 250 pixel dall'inizio pagina. Inoltre il dropdown del login è gestito interamente da javascript, sia l'apparizione che il movimento dalla sezione principale alle due di login e registrazione, e anche la freccia di navigazione che permette di tornare indietro.

In alcune pagine, il caricamento dei dati (o l'invio di essi) non avviene tramite una richiesta get/post da una form, bensì tramite una richiesta AJAX al server. Questo rende più responsive le pagine che ne hanno bisogno, caricando i dati necessari immediatamente dopo l'input utente e senza bisogno

del refresh della pagina. Per esempio si vedano le pagine Candidature, Proiezioni, Lista Film della sezione Admin.

Il sito è ben usabile anche con javascript disattivato. Non sono possibili le funzioni di login, acquisto biglietti, candidatura al CPContest e la sezione admin. Tutto il resto del sito è consultabile.

Il javascript usato è compatibile con versioni molto vecchie dei browser (Edge 12, Firefox 09, Opera 12.1, Chrome 8, Safari 6, IE10) per quanto riguarda la parte non admin. Per la parte admin invece i requisiti sono un po' più alti mentre Internet Explorer non è utilizzabile (Edge 17, Firefox 29, Opera 38, Chrome 51, Safari 10).

Questa ampia compatibilità è anche grazie al fatto che alcune funzioni che alzavano sostanzialmente i requisiti siano state implementate in un particolare modo:

- **Passive event listeners:** aumentano di molto l'efficacia di scrolling della pagina in presenza di eventi onscroll, tuttavia non sono supportati da tutti i browser. Pertanto si è deciso di effettuare una **feature detection** all'inizio dello script per verificare la presenza di questa feature.
- **String.prototype.includes:** funzione semplice che però non è stata implementata velocemente dai browser, si è deciso di implementarla (qualora fosse assente), tramite una funzione sostitutiva che usa String.prototype.indexOf per implementarla. Questa tecnica è detta **polyfill**.
- **Input type="date" in Safari < 14.1:** Come verrà descritto sotto nella sezione dei Browser, Safari fino a versione molto recente non implementa questo tipo di input, mostrandolo come se fosse un campo di testo, pertanto si è scelto di aggiungere un placeholder in questo campo per aiutare l'utente a inserire la data nel formato corretto, esso viene aggiunto solo in Safari, per evitare errori di validazione (l'attributo placeholder non può normalmente stare in un input type="date").

È presente un tasto "Torna su" gestito con javascript, esso viene nascosto quando l'utente è all'inizio pagina e mostrato quando l'utente comincia a scrollare oltre una certa threshold.

Per compatibilità si è scelto di lasciare tale freccia visibile di default, in quanto nel caso di javascript disattivato essa si veda lo stesso. Questo comportamento comporta un piccolo glitch grafico quando una pagina viene caricata, infatti si vede la freccia e subito dopo essa viene nascosta, questo usando javascript è inevitabile (almeno per le nostre conoscenze), si è in ogni caso valutato che i benefici di tale scelta ne valgono la pena.

Comportamento Php

Il codice Php è organizzato principalmente nelle seguenti entità:

- **Class DBAccess:** gestisce la connessione con il DB, formatta i risultati delle query e contiene una funzione per query usata nel resto del codice.
- **Class FS:** gestisce l'inserimento e la ricerca delle immagini, ogni film ne impiega una.
- **Class Login:** gestisce le operazioni di Login, Logout e Registrazione degli utenti.
- **Class Sanitizer:** fornisce delle funzioni usate nella sanificazione dell'output contenuto nel DB e stampato a schermo, impedisce l'esecuzione di codice malevolo introdotto nel DB.
- **Class Utils:** fornisce delle funzioni utili alla validazione lato server, per l'inserimento di feedback a schermo dopo una richiesta POST e per la rimozione di *placeholder* non utilizzati.
- **Pagine php:** Ogni pagina del sito è gestita dal proprio file php; esso definisce i metodi POST e GET e il flow logico del programma.

Pagine php

Ogni pagina php implementa la stessa struttura, si differenziano le richieste POST, le richieste GET e le funzioni di stampa del feedback.

Per gestire le richieste **POST** si effettua un controllo alla variabile *“method”* presente in ogni tag `input type=“submit”`, leggendo il *“value”* della stessa e chiamando le opportune funzioni php. Le funzioni POST prevedono l'esecuzione delle query e la definizione delle variabili di sessione *method*, che ricorda la richiesta effettuata, *success*, che definisce il risultato positivo o negativo della query, e *feedback*, che contiene il messaggio di feedback da fornire all'utente. Quando è previsto l'inserimento di dati da parte dell'utente viene predisposta la validazione di essi utilizzando *Utils::validate*, inoltre se la richiesta ha esito negativo i dati inseriti dall'utente vengono salvati e stampati a schermo nuovamente, assieme al messaggio di errore, così l'utente può rimediare al suo errore senza dover ricompilare la form da zero.

Alla fine di una richiesta **POST** viene effettuato un redirect tramite il response header 303, che reindirizza alla pagina stessa. Questo per evitare comportamenti non desiderabili legati al refresh della pagina nella quale si è mandata una richiesta **POST** e al salvataggio di essa nei bookmarks. Per esempio, se si comprasse un biglietto tramite una **POST** basterebbe fare il refresh accidentale della pagina per comprarne un altro. Questo si evita appunto tramite il paradigma **POST-redirect-GET**.

Successivamente vengono eseguite le funzioni **GET** che gestiscono la stampa del codice HTML usando i risultati delle relative query. Qualora sia prevista la stampa di dati presenti nel DB si fa uso dell'opportuna funzione di sanificazione.

Si è scelto di effettuare la stampa dei contenuti dinamici del sito tramite il replace di alcuni placeholder, questa scelta è motivata dalla volontà di tenere separato l'html dal php, così l'html viene caricato e modificato dal PHP in autonomia, consentendo di scrivere l'HTML senza preoccuparsi dell'implementazione del backend. In particolare, grazie a questo paradigma il PHP rende disponibili alcuni tag che risultano molto utili ed efficaci nella gestione dell'HTML, ossia (tra le altre queste sono le più rilevanti e usate) le due sezioni *“<iflogged> ... </iflogged>”* e *“<ifnotlogged> ... </ifnotlogged>”* le quali, come si evince dal nome, vengono rispettivamente mostrate e nascoste dal php se l'utente che visita la pagina è loggato, e viceversa. La stessa cosa accade anche con gli account admin con *“<ifloggedadmin> ... ecc.”*.

Infine, si gestiscono le funzioni di **feedback**. Esse vengono chiamate quando è stata invocata una richiesta POST e si controllano quindi le variabili della sessione impostati nel corpo delle funzioni che gestiscono le suddette richieste. Si identifica la richiesta invocata tramite la variabile *method* e viene usata *Utils::printFeedback* con l'opportuno tag placeholder nella pagina HTML. Dopo aver effettuato la stampa si effettua un unset delle variabili di sessione usate in precedenza e una chiamata a *Utils::feedbackcleanUp* per cancellare i placeholder non usati.

La scelta di usare php per la gestione di ogni pagina, anche di quelle statiche, è influenzata dall'implementazione delle funzioni di Login/Signup in ogni pagina del sito.

Ricapitolando, il paradigma **POST-redirect-GET** impone che per prima cosa si gestisce una richiesta **POST** e si salvano i dati e l'esito della richiesta nelle variabili di sessione, poi si manda una risposta di redirect che causa al client di mandare una richiesta **GET** alla pagina dove è stato reindirizzato, e in questa richiesta **GET** il server stampa eventuali feedback o dati.

La gestione delle query del database è fatta tramite la libreria mysql offerta da PHP, in particolare per ogni query che involve dati dinamici inseriti dall'utente si usano le cosiddette **query prepare**.

Esse garantiscono che i dati inseriti siano inviati completamente separati dalla stringa di query SQL e garantiscono **immunità all'SQL injection** senza il bisogno di sanificare in modi imprecisi l'input dell'utente. La sanificazione, infatti, avviene solo quando si stampano nel browser i dati provenienti dal database, considerati non sicuri perché inseriti dall'utente, tramite le funzioni presenti nella classe Sanitizer.

4. Accessibilità

Fin dai primi momenti di realizzazione, l'accessibilità è stata parte delle scelte implementative, come dovrebbe essere. Non è stata quindi una questione di "correzione a posteriori", seppure eseguendo gli opportuni test e conseguenti correzioni a sito terminato.

Presa in considerazione la bassa partecipazione a eventi collegati al cinema di utenti non vedenti, ipovedenti, o con problemi legati alla vista, ciò non significa che non possano navigare il sito, acquistare biglietti e venire alle proiezioni. Si è cercato di rendere il sito accessibile anche da questo punto di vista.

Alcune cose di cui abbiamo tenuto conto:

- **Html corretto:** Usando i corretti tag Html, in modo semanticamente corretto e nel corretto ordine, si crea una pagina ben strutturata e chiara. Ciò aiuta non solo gli screen-reader, ma anche i browser, la manutenzione, la navigazione con tab, eccetera...
- **La lingua delle parole.** Usando l'attributo "lang" nel codice html, si segnala la lingua delle parole non italiane. (Home, About Us...)
- **Abbreviazioni:** Usando il tag "abbr" si segnalano le parole che sono abbreviazioni e si scrive la parola completa. (ecc=eccetera, CP=CinePado)
- **Testo chiaro:** Scrivere un contenuto che sia chiaro, ma soprattutto usare un font semplice da leggere, senza grazie, usare una buona interlinea e suddividere in paragrafi.
- **Considerare tutti i tipi di utenza:** Si è cercato di farlo in tutto il sito, ma in particolare nella pagina "Come Trovarci" e nella sezione "Accessibilità" di "About Us", un occhio di riguardo è dedicato alle persone con disabilità varie.
- **Form:** Sono stati usati i tag "label", i feedback di successo e di errore.
- **Non usare link circolari,** ossia che rimandano alla pagina in uso.

Link

I link all'interno del testo sono ben segnalati: sono sottolineati e colorati in blu. Se portano a pagine già visitate, il colore è viola. Anche i link ai film nelle pagine "Film" e "Proiezioni" sono sottolineati e cambiano colore se la pagina è già stata visitata.

Seguono la convenzione anche il menu nell'header, il testo sopra le immagini alla fine della home page e l'acquisto biglietti in base al giorno, presente nella home page. In questo ultimo caso è molto utile per l'utente sapere se ha già guardato le proiezioni di un certo giorno, quindi anche qui i link alle pagine già visitate sono in un altro colore.

Altri link all'interno del sito sono facilmente riconoscibili perché mostrati come bottoni o icone, il testo non è quindi sottolineato. Questi link non seguono la convenzione del cambio colore se la pagina è già visitata, in quanto è stato valutato che non darebbe molto aiuto all'utente, quanto piuttosto l'alternarsi di colori creerebbe confusione e renderebbe meno piacevole l'uso della pagina.

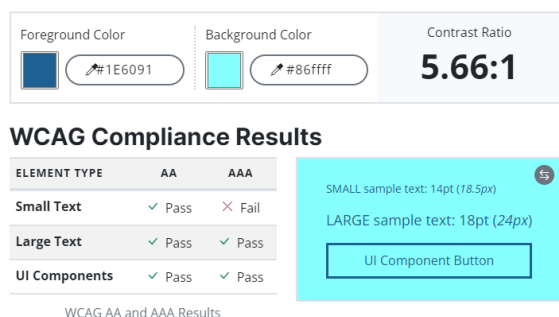
Colori

Il contrasto tra i vari colori è stato controllato usando gli strumenti per sviluppatori di Chrome. Per i colori con contrasti minore di 1:6 è stato fatto un ulteriore controllo su diversi siti (tra cui <https://accessibleweb.com/color-contrast-checker/>, <https://contrastchecker.online/>, <https://color.a11y.com/>). Di seguito la lista dei colori con rapporto minore di 6 controllati in più siti, e due screenshot di esempio:

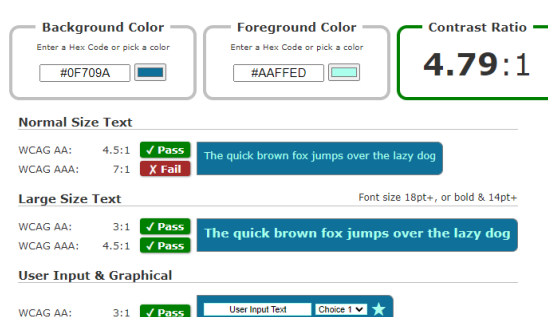
Menu nav visitato non hover, alcuni bottoni, link nel testo non visitati (#FFFFFF su #0F709A): 5.51, menu nav non visitato non hover (#AAFFED su #0F709A): 4.79, acquisto biglietto per giorno se visitato, in home (#800080 su #74D9DD): 5.7, link non visitato in breadcrumb (#0000EE su #168AAD77): 5.31, bottone acquista nella pagina di proiezioni (#1E6091 su #86FFFF): 5.66, link non visitato nel menu admin (#0000EE su #74d9dd): 5.92.

Tutti i contrasti rispettano almeno le norme WCAG AA.

Bottone acquista nella pagina "Proiezioni"



Menu nav non visitato



Nella realizzazione del sito sono state apportate alcune accortezze riguardo i colori. Dove i colori veicolano informazione, ad esempio nel feedback di errore delle form, si è aggiunto il grassetto. Non sono stati fatti riferimenti a colori (niente “clicca sul pulsante blu”).

Immagini

Essendo un sito collegato al cinema, la componente visiva e di immagini è importante.

Ad ogni film è associata un’immagine e ad ogni immagine è associata una descrizione testuale, che racconta ciò che si vede dentro l’inquadratura. Questa descrizione testuale è inserita dall’admin e salvata nel database.

Oltre a questo, si è cercato però di non veicolare troppo contenuto attraverso le immagini. La tecnica di image replacement è stata usata per il logo nell’header e i loghi dei social nella sezione “social” della pagina “About Us”.

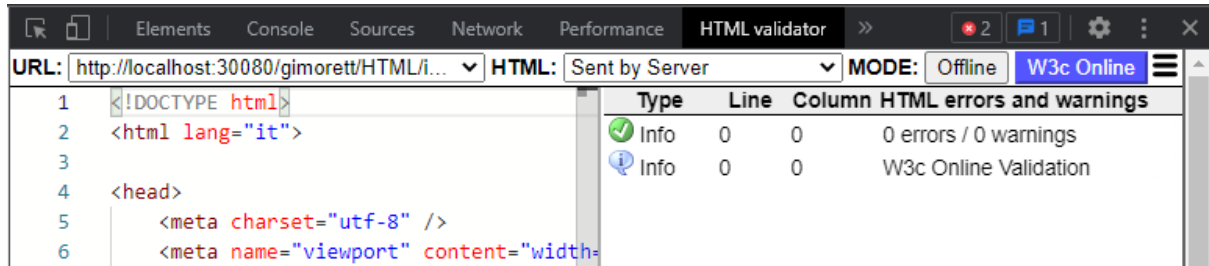
Le uniche altre immagini presenti nel sito sono i loghi degli sponsor e le tre foto in fondo alla home, tutte con alternativa testuale coerente.

Test

Validazione

Per poter validare il codice HTML dopo l’esecuzione dei file php, è stata utilizzata l’estensione “[HTML Validator](#)”, con MODE: W3c online, che permette di mandare il codice al validatore del w3c.

Validazione pagina home



I file css sono invece stati validati per input diretto tramite il [“Nu Html Checker”](#) del w3c.

Screen Reader

Il sito è stato testato esaurientemente con uno screen reader (NVDA), sono stati aggiunti tutti gli attributi aria necessari da renderlo accessibile e usabile da utenti disabili.

Il sito si presentava già accessibile grazie all’uso di tag html adeguati, sono state pertanto aggiunte label ai menù, descrizioni di varie liste di elementi, e alcune informazioni bonus per dare contesto ad un utente che non può vedere lo schermo, come (per esempio) una label introduttiva ad ogni “” della lista dei film e delle proiezioni. Inoltre i messaggi di errore dinamici lato client sono resi accessibili segnalando tramite attributi aria che un campo non è valido, e fornendo il messaggio di errore come descrizione dello stesso, evitando ambiguità e rendendo facile capire se un campo è sbagliato e in tal caso cosa c’è di sbagliato in esso.

L’unica parte non accessibile del sito era il dialogo del login, che è stato reso accessibile grazie alla gestione dinamica del focus e agli attributi aria che descrivono la relazione tra il dialogo e il bottone all’interno dell’header che lo apre.

Focus

La gestione del focus è molto importante per l’accessibilità, in quanto aiuta gli utenti di varie piattaforme prive di puntatori precisi, come utenti da tastiera, command line, screen reader, e in alcuni casi aiuta anche gli utenti mobile.

Oltre alla gestione del focus nel dialogo di login, il focus viene spostato nel luogo adeguato ogni volta che si interagisce con un elemento a comparsa, come si può vedere nelle pagine di Admin. Inoltre, ogni volta che c’è un feedback lato server esso viene messo in focus di default, così da essere letto immediatamente dopo il load da screen reader o altri dispositivi.

Browser

Il sito e le sue funzionalità sono state testate sui seguenti browser, i più utilizzati in Italia, secondo StatCounter, a dicembre 2021, considerando tutti i tipi di device:

- Chrome (utilizzato dal 66,72%): Browser principale usato nella fase di sviluppo del sito, quindi ogni cosa funziona correttamente. La presentazione segue precisamente le volontà espresse nella realizzazione del sito.
- Safari (utilizzato dal 18,62%): Browser molto usato, funziona correttamente nelle ultime versioni, e la presentazione è coerente agli altri browser. Abbiamo però verificato che ci sia la compatibilità, riscontrando un problema principale: il tag input supporta il type=”date” solo dalla versione di Safari 14.1 (maggio 2021), da noi valutata troppo recente. Questo tag è usato per inserire la data di nascita nelle form per la registrazione di un utente. Safari in alternativa mostra un input di testo, ma l’utente così non è a conoscenza del formato per

inserire la data. Per risolvere questo problema, viene inserito tramite javascript un placeholder nel value dell'input (come per gli altri input di testo) che mostri il formato di inserimento della data, solo nel caso in cui il browser utilizzato sia Safari. Questo è stato testato con successo in un Safari 13.1.

- Firefox (utilizzato dal 4,61%): Funzionamento corretto, presentazione coerente.
- Edge (utilizzato dal 3,91%): Funzionamento corretto, presentazione coerente.
- Samsung Internet (utilizzato dal 3,49%): Browser per mobile, funzionamento corretto, presentazione coerente.
- Opera (utilizzato dal 1,45%): Funzionamento corretto, presentazione coerente.
- Internet Explorer (utilizzato dal 0,43%): Il sito è consultabile, seppur con qualche difetto nel layout di alcuni elementi. Le funzionalità che usano Javascript (login e registrazione, acquisto biglietti) non sono utilizzabili.

Dispositivi

Il sito e le sue funzionalità sono state testate su pc desktop, smartphone e tablet. In particolare, oltre ad utilizzare gli strumenti per sviluppatori dei vari browser, il sito è stato testato direttamente su vari dispositivi (smartphone e tablet) per verificarne l'usabilità.

Strumenti Accessibilità

- Silktide (estensione chrome): Usato per simulare dislessia, miopia e soprattutto daltonismo. In particolare si è notato che sono distinguibili tra loro i colori del testo, dei link visitati e dei link non visitati anche con le forme più gravi e rare di daltonismo.
- Text zoom di Abilitools: Aumentando la dimensione del testo tramite questa estensione di Chrome, ci si è resi conto che alcune parti, ad esempio la sezione "acquisto biglietti" in home, non si trasformano in modo piacevole e diventano poco leggibili. Questo problema non è stato risolto, perché accade solo se viene aumentata la sola dimensione del testo (e di molto). In caso di zoom della pagina intera (il classico ctrl+rotellina mouse) la trasformazione è elegante e priva di problemi. Inoltre, tra le persone con problemi alla vista è più comune lo strumento di zoom dello schermo intero, e non il solo aumento della dimensione del font.
- Wave (estensione chrome): Fornisce un utile feedback visivo sull'accessibilità della pagina. Oltre ad un trascurabile errore di contrasto nel logo (lampadine gialle su scritta bianca), ha mostrato alcuni warning interessanti. Ad esempio: link ridondanti vicini (l'header con il logo rimanda alla home, così come la scritta "Home" nella barra del menu), jump menu con javascript (nella sezione admin, in più punti la selezione da un tag <select> porta modifiche nella pagina). Dopo una valutazione di queste due questioni, si è deciso per entrambe di non mantenere così come è stato inizialmente implementato. In particolare la questione del "jump menu": visto il contesto in cui è utilizzato, si è deciso che anche con l'utilizzo da tastiera è molto più comodo il "jump menu", e la modifica automatica della pagina è chiara e senza bisogno di essere notificata. L'aggiunta di un bottone di "conferma della selezione" è stata valutata controproducente.

Errori

Alcune tool automatiche presentano degli errori classificati come "falsi positivi", i casi sono documentati qui assieme a una spiegazione per la quale l'errore non è corretto:

- "select has non accessible name": ogni select ha una label ed è stata testata essere accessibile, sia tramite ispezione nella tab "accessibilità" di edge/chrome, sia tramite screen reader NVDA.

- “Aria attribute not allowed (aria-expanded)”: dalle informazioni ufficiali invece è chiaro si possa usare tale attributo sui checkbox, in ogni caso funziona nel modo corretto ed è inteso proprio per l'uso che ne viene fatto, come da documentazione ufficiale.
- “Input type='checkbox' missing required attribute role”: in realtà il ruolo di tale elemento è implicito e dato da html5, infatti un input type='checkbox' ha sempre role="checkbox" implicitamente, come verificato tramite ispezione e da screen reader

6. Organizzazione del lavoro

Suddivisione interna dei compiti:

- Struttura HTML: Tommaso Di Fant, Alberto Nicoletti, Luca Annicchiarico, Giovanni Moretti
- CSS desktop: Tommaso Di Fant, Alberto Nicoletti, Luca Annicchiarico, Giovanni Moretti
- CSS mobile: Giovanni Moretti
- CSS stampa: Alberto Nicoletti
- Progettazione Database: Tommaso Di Fant, Luca Annicchiarico
- Php: Tommaso Di Fant, Luca Annicchiarico
- Javascript: Tommaso Di Fant, Alberto Nicoletti, Luca Annicchiarico
- Accessibilità: Tommaso Di Fant, Alberto Nicoletti
- Contenuti: Alberto Nicoletti
- Testing: Alberto Nicoletti, Luca Annicchiarico, Giovanni Moretti
- Relazione: Tommaso Di Fant, Alberto Nicoletti, Luca Annicchiarico, Giovanni Moretti