

UNIVERSITÀ DEGLI STUDI DI PADOVA

CORSO DI LAUREA IN INFORMATICA
PROGETTO DI TECNOLOGIE WEB

DevSpace

A.A. 2018/2019

Relazione

<i>Team:</i>	<i>Matricola:</i>
Giacomo BARZON	1143164
Francesco DE FILIPPIS	1143408
Giacomo GREGGIO	1142951
Michele ROVERATO	1143030

Informazioni logistiche

Il sito è accessibile al seguente link:

<http://testsitotecweb.altervista.org/>

Account non amministratore utilizzabile:

- **Username:** linus@torvalds.com
- **Password:** git

Account amministratore utilizzabile:

- **Username:** admin@admin.it
- **Password:** admin

Indice

1	Analisi dei requisiti	4
1.1	Requisiti	4
1.2	Target di utenza	4
2	Architettura delle informazioni	4
2.1	Descrizione	4
2.2	Struttura pagine pubbliche	4
2.3	Struttura pagine privata	6
3	Usabilità	7
3.1	Introduzione	7
3.2	Layout e design	7
3.2.1	Scroll verticale	7
3.3	Favicon	7
3.4	Gestione dell'errore 404	7
3.5	Sidebar	8
3.6	Visualizzazione sotto-argomenti e relativi articoli	8
3.7	Minificazioni	8
3.8	Media queries	8
4	Accessibilità	9
4.1	Introduzione	9
4.2	Scelte adottate	9
4.2.1	Colori	9
4.2.2	Link	9
4.2.3	JavaScript	9
4.2.4	Breadcrumb	10
4.2.5	Navigazione tramite Tab e accessKey	10
5	Sviluppo	10
5.1	Tecnologie utilizzate	10
5.2	Strumenti di supporto	11
5.3	Form	11
5.3.1	Struttura dei form	11
5.3.2	Validazione dei form	11
5.3.3	Validazione HTML5	11
5.3.4	Validazione JavaScript	11
5.3.5	Validazione PHP	12
5.3.6	Scrittura articoli	12
5.4	Separazione parte privata e pubblica	12
5.5	Strumenti amministratore	13
5.6	Classi	13
5.7	Gestione degli errori	14

5.8	Progettazione del Database	15
5.9	Gestione Javascript	16
5.10	Ricerca dei contenuti	16
5.11	Menù e sidebar senza JavaScript	17
5.12	Framework scrollbar	17
6	Compatibilità	18
6.1	Descrizione	18
6.2	Tag HTML5 utilizzati	18
6.3	Browser supportati	18
6.4	JavaScript	19
7	Test	19
7.1	Accessibilità	19
7.1.1	Strumenti utilizzati	19
7.1.2	Risultati	20
7.2	Usabilità	21
7.2.1	Strumenti utilizzati	21
7.2.2	Risultati	21
7.3	Performance	21
7.3.1	Strumenti utilizzati	21
7.3.2	Risultati	21
7.4	Validazione	22
7.4.1	Strumenti utilizzati	22
7.4.2	Risultati	22
8	Suddivisione dei ruoli	22

1 Analisi dei requisiti

1.1 Requisiti

Lo scopo di questo progetto consiste nel dare vita ad una piattaforma web che offre la possibilità di consultare un archivio di informazioni sugli argomenti del mondo informatico. Gli utenti troveranno le informazioni suddivise nel seguente modo:

- Nella home page sono disponibili delle card che identificano il macroargomento, mostrano una immagine ad esso relativa e vengono elencati i principali sottoargomenti tramite dei link;
- I sottoargomenti, a cui è dedicata una pagina, contengono dei link che si riferiscono agli articoli facenti parte di quel particolare sottoargomento.

L'utente ha inoltre la possibilità di commentare i singoli articoli, interagendo così con gli altri utenti registrati al blog. L'utente ha la possibilità di lasciare un giudizio(pollice in sù/pollice verso) ai commenti degli altri utenti. Gli amministratori dispongono di un pannello amministratore in cui vengono elencate le operazioni che possono compiere, tra cui aggiornare la piattaforma con nuovi contenuti(eventualmente migliorando contenuti già esistenti), sospendere un account utente e nominare nuovi amministratori. Devono inoltre occuparsi di moderare l'ambiente sfruttando la possibilità di eliminare eventuali commenti inopportuni.

1.2 Target di utenza

Il sito è rivolto ad un pubblico che ha dimestichezza con la tecnologia e certamente con la ricerca e l'utilizzo di Internet per trovare informazioni. Inoltre può essere consultato da persone che desiderano acquisire delle nozioni sui principali argomenti riguardanti l'informatica, perciò si prevede che la piattaforma verrà raggiunta da utenti con età superiore ai 14 anni (e.g. dagli studi superiori in poi).

2 Architettura delle informazioni

2.1 Descrizione

In questa sezione verrà descritto il modo in cui è stato scelto di gestire le informazioni presentate nel sito. Poiché si tratta di un blog in cui la quantità di contenuti è destinata a crescere nel tempo è stata adottata un'interfaccia a tre pannelli, navbar, sidebar e contenuto. La spiegazione dettagliata verrà fornita nella **Sezione Usabilità**.

2.2 Struttura pagine pubbliche

La parte pubblica del sito è accessibile a tutti gli utenti e di seguito verranno descritte le pagine visitabili con le relative funzionalità offerte.

- **Homepage (index.php)**: si tratta della pagina principale del sito in cui l'utente trova le seguenti informazioni:
 - **Navbar**: parte superiore della pagina in cui viene mostrato il logo del sito e una lista orizzontale di link utili alla navigazione;
 - **Header**: sezione sottostante alla navbar composta da immagine di sfondo e nome del sito;
 - **Barra di ricerca**: permette la ricerca di contenuti specifici che riguardano macroargomenti, sottoargomenti e articoli;
 - **Descrizione e Contenuto**: nella parte inferiore dell'header è presente una descrizione del sito, seguita dalla lista dei macroargomenti visualizzati sottoforma di card;
 - **Footer**: parte inferiore del sito contenente ulteriori informazioni sulla piattaforma.
- **About (about.php)**: è una pagina che fornisce tutte le informazioni sul sito, ovvero lo scopo dello stesso e gli sviluppatori che lo hanno realizzato. Anche in questa pagina è presente la navbar per la navigabilità, l'header e il footer;
- **Ricerca (search.php)**: in questa pagina vengono visualizzati i risultati di una ricerca avvenuta tramite la casella di ricerca, utile per trovare in modo veloce specifici argomenti e/o articoli;
- **Registrazione registrazione.php**: pagina che offre la possibilità di creare un account agli utenti non registrati al blog;
- **Login login.php**: pagina che offre la possibilità di effettuare il login agli utenti che sono registrati alla piattaforma;
- **Link Articoli (ArticleLinks.php)**: questa pagina contiene tutti i sottoargomenti, con i relativi link agli articoli, appartenenti ognuno al proprio macroargomento. La pagina è strutturata nel seguente modo:
 - Sulla sinistra è presente una sidebar che elenca tutti i macroargomenti;
 - Nel contenuto, a inizio pagina, a seguito della descrizione del macroargomento che è stato selezionato, vengono elencate tramite link le ancore ai sottoargomenti presenti;
 - Per ciascun sottoargomento, vengono elencati i link degli articoli relativi;
 - Nella parte superiore è presente la navbar già descritta precedentemente, e nella parte inferiore è situato il footer, anch'esso già descritto.
- **Articolo (ReadArticle.php)**: questa pagina è strutturata nello stesso modo di quella descritta al punto precedente, ma nel contenuto vi è il corpo di un articolo che l'utente ha raggiunto utilizzando uno dei link elencati sotto ciascun sottoargomento della precedente pagina;

- **Account (profile.php)**: in questa pagina l'utente può visualizzare le sue informazioni personali, ed eventualmente le può modificare;

2.3 Struttura pagine privata

La parte privata del sito è accessibile ai soli account amministratori della piattaforma. Di seguito verranno descritte le pagine di interesse con le relative funzionalità offerte.

- **Link Articoli (ArticleLinks.php)**: questa pagina, già descritta, presenta un aspetto diverso per l'utente amministratore, il quale può:
 - Aggiungere un nuovo sottoargomento;
 - Eliminare un sottoargomento esistente;
 - Scrivere un nuovo articolo;
 - Modificare un articolo esistente;
 - Eliminare un articolo esistente.
- **Articolo (ReadArticle.php)**: la pagina, già descritta nella sezione precedente, permette ad un amministratore di eliminare i commenti degli utenti, nel caso in cui non siano pertinenti;
- **Scrivi Articolo (writeArticle.php)**: questa pagina è dedicata alla gestione di un singolo articolo, in particolare l'amministrare può modificare il contenuto di un articolo già esistente, oppure scriverne uno nuovo;
- **Strumenti (adminTools.php)**: in questa pagina l'utente amministratore visualizza una lista delle operazioni che può compiere, ovvero:
 - Creare un nuovo macroargomento;
 - Aggiungere un nuovo amministratore;
 - Gestire utenti sospesi.
- **Gestione argomenti (manageArguments.php)**: questa pagina permette all'amministratore di creare un nuovo macroargomento e di eliminare quelli esistenti;
- **Aggiunta amministratore (addAdmin.php)**: in questa pagina un utente amministratore può aggiungere un nuovo utente al gruppo dei moderatori del sito fornendo il suo indirizzo email;
- **Sospensione account (manageUsers.php)**: in questa pagina un utente amministratore può sospendere un utente, restringendo così le sue attività sulla piattaforma, oppure revocare la sospensione ad utenti attualmente sospesi.

3 Usabilità

3.1 Introduzione

Essendo il sito rivolto ad un pubblico potenzialmente vario, è stato adottato lo schema a tre pannelli il quale, a nostro parere, permette una buona navigabilità data la natura del sito.

3.2 Layout e design

Tutte le pagine che presentano i contenuti, quindi presumibilmente le più visitate, sono dotate di una sidebar che offre un modo intuitivo e veloce per navigare attraverso macroargomenti, sottoargomenti, una navbar che permette di spostarsi in altre aree del sito con un solo click e un footer che fornisce informazioni aggiuntive sul sito.

3.2.1 Scroll verticale

Poiché il sito è destinato alla raccolta di contenuti, è possibile che nel tempo questi crescano, generando così delle pagine o sezioni potenzialmente lunghe (soprattutto la sidebar e la pagina con i link agli articoli). Su mobile inoltre, questo tipo di situazione è sentita maggiormente date le dimensioni degli schermi dei dispositivi mobili. L'unico accorgimento per ovviare a questo problema, non tanto per lo scroll necessario alla discesa della pagina quanto alla risalita, è stato l'introduzione di un pulsante che permetta di tornare all'inizio della pagina. Tuttavia esistono oggi moltissimi siti e piattaforme che fanno uso di strutture simili per la presentazione di una grande mole di contenuti, perciò riteniamo che l'utente medio giunto sul nostro sito trovi familiare e semplice questo tipo di layout e modalità di navigazione.

3.3 Favicon

Tutte le pagine sono dotate di favicon che permettono la facile individuazione del sito tra le schede del browser o nei preferiti. Per generare le favicon in diversi formati è stato utilizzato il tool offerto al seguente link:

<https://www.favicon-generator.org/>

3.4 Gestione dell'errore 404

È stato previsto un apposito reindirizzamento a una pagina di errore nel caso di richiesta di reperimento, da parte dell'utente, di una pagina non esistente. All'interno di questa pagina di errore 404 è stato inserito un apposito messaggio di errore comprendente un link per poter tornare alla home page.

3.5 Sidebar

La sidebar come già accennato è pensata per offrire un modo semplice ed intuitivo per navigare tra i contenuti del sito, pertanto costituisce un elemento cruciale all'interno della nostra piattaforma. Nella sidebar è presente per prima cosa una casella di ricerca, la quale permette di cercare immediatamente dei contenuti specifici invece di navigare per le varie voci presenti più in basso. Sotto la casella di ricerca vi è una lista di voci che rappresentano i macroargomenti, e ciascuna di queste voci si può espandere per visualizzare i rispettivi sottoargomenti, i quali sono link ancora verso la parte centrale della pagina che mostra i sottoargomenti con relativi articoli. Per facilitare l'orientamento dell'utente abbiamo fatto uso dei *breadcrumb* che si occupano di evidenziare il macroargomento (e la lista dei sottoargomenti) che si sta visualizzando oppure nel caso di un articolo, viene evidenziato il sottoargomento a cui appartiene l'articolo che si sta leggendo.

3.6 Visualizzazione sotto-argomenti e relativi articoli

Per minimizzare il numero di click che l'utente deve effettuare per raggiungere un articolo, il quale non dovrebbe essere superiore a 3 o 4, si è deciso di visualizzare all'interno di un'unica pagina tutti i sotto-argomenti ed i relativi articoli ad essi associati e poi, sia nella sidebar che nelle card presenti all'interno della home, elencare per ogni argomento delle ancore che puntano al sotto-argomento specifico nella pagina descritta precedentemente. L'alternativa sarebbe stata quella di creare due pagine separate per argomenti e sotto-argomenti, tuttavia abbiamo ritenuto che questa scelta avrebbe rallentato notevolmente la navigazione aumentando considerevolmente il numero di click necessari per spostarsi da articolo ad articolo. Inoltre riteniamo che l'implementazione da noi effettuata non generi problemi di usabilità né problemi di accessibilità.

3.7 Minificazioni

I file .css e .js generalmente vengono minificati al fine di ridurre il loro peso complessivo e quindi velocizzare la richiesta della risorsa al server. Per questioni di leggibilità non è stato fatto, eccetto per le librerie JavaScript che abbiamo utilizzato per gestire lo stile della scrollbar situata nella sidebar poiché non sviluppate da noi.

3.8 Media queries

Per garantire la stessa esperienza su tutti i dispositivi si è fatto uso di media queries, in particolare abbiamo individuato due *breakpoint* che ci hanno permesso di adottare una sufficiente suddivisione delle regole css relative ad ampie categorie di dispositivi differenti; 1276px e 768px. Per quanto riguarda la possibilità di stampare le pagine del sito, abbiamo scritto un foglio di stile *print.css* che, in fase di stampa, riadatta i contenuti laddove possibile e si occupa di rimuovere gli elementi di non utilità per un documento su carta, per esempio navbar e sidebar in quanto elementi prettamente di navigabilità più che di informazione.

4 Accessibilità

4.1 Introduzione

Affinché il sito raggiunga la vasta gamma di utenti per cui è stato progettato, particolare attenzione è stata posta sul fattore accessibilità seguendo i principi fondamentali descritti dal WCAG 2.0. In particolare, nel nostro caso specifico, sono stati adottati gli accorgimenti di seguito descritti.

4.2 Scelte adottate

In questa sezione verranno illustrate le scelte intraprese al fine di rendere il sito accessibile.

4.2.1 Colori

Per quanto riguarda l'utilizzo dei colori per i testi e gli elementi di tutte le pagine, abbiamo assicurato di ottenere un rapporto di contrasto pari 7:1 tra un elemento in foreground e uno in background, rendendo così il sito conforme ai livelli di accessibilità AA e AAA del WCAG 2.0.

4.2.2 Link

Sono stati uniformati i link presenti all'interno del sito web, in modo da renderli facilmente riconoscibili da qualsiasi tipologia di utente. La maggior parte dei link è facilmente riconoscibile da una particolare tonalità di blu, e dalla caratteristica sottolineatura diventata ormai uno standard all'interno del web. Gli unici link che fanno eccezione a queste regole sono quelli presenti all'interno della sidebar per questioni estetiche e di contrasto. Il team ritiene tuttavia che siano ugualmente riconoscibili grazie alla notevole diffusione all'interno del web di sidebar dalla struttura simile. Come si può notare navigando tra le pagine del sito, non è stato previsto la gestione dei link visited perché secondo il team era sufficiente l'utilizzo dei breadcrumb sia nel menù situato sulla navbar (che ha poche voci e molto basilari) sia nella sidebar (anch'essa dotata di breadcrumb) poiché essendo quest'ultima utilizzata per navigare i contenuti del sito l'utente potrebbe, dopo poco tempo di navigazione, aver visitato molte delle voci (o quasi tutte) rendendo poco utile la funzione svolta dai link visited.

4.2.3 JavaScript

Il sito risponde bene in caso di assenza di JavaScript in quanto non è stato utilizzato per alterare la struttura della pagina, ma solo per offrire una migliore esperienza d'uso di alcuni elementi quali il menù a tendina, la sidebar e la scrollbar associata alla sidebar. Inoltre è stato utilizzato per i controlli dei form client-side e per mostrare messaggi di errori con un miglior aspetto. Perciò il caso in cui JavaScript non fosse disponibile, viene preclusa la possibilità di utilizzare il sito e consultarne i contenuti. Nella sezione 5 (Sviluppo) verranno illustrate, in diverse sezioni, le tecniche adottate per gestire le funzionalità appena descritte qualora JavaScript fosse appunto disabilitato oppure non presente.

4.2.4 Breadcrumb

Abbiamo utilizzato i breadcrumb per due differenti aree e funzionalità del sito. La prima riguarda la navbar, presente in tutte le pagine, che contiene i link per navigare in tutto il sito. Le voci del menù vengono evidenziate e i link disattivati in modo congruo alla pagina in cui l'utente si trova in quel preciso momento. Il secondo utilizzo di breadcrumb viene fatto nella sidebar, dove viene evidenziato l'elenco corrispondente al macroargomento e relativi sottoargomenti che l'utente sta visualizzando e, nel caso stesse leggendo un articolo, viene evidenziato solo il sottoargomento a cui esso appartiene.

4.2.5 Navigazione tramite Tab e accessKey

Il sito è stato realizzato in modo tale che fosse prevista la navigazione anche senza la presenza di un mouse e dunque tramite esclusivo utilizzo della tastiera. E' stato dunque ottimizzato al fine di renderlo navigabile tramite "Tab" e sono state inserite delle shortcut grazie all'uso dell'attributo accessKey per raggiungere velocemente le sezioni principali del sito. Queste shortcut riguardano principalmente la navbar e sono elencate qui di seguito:

- **h**: per il riferimento alla home page;
- **l**: per il login/logout;
- **a**: per la pagina about;
- **c**: per creare un nuovo account(registrazione);
- **p**: per il profilo utente;
- **t**: per i tools(strumenti) amministratore;

5 Sviluppo

5.1 Tecnologie utilizzate

Le tecnologie utilizzate per sviluppare il sito sono le seguenti:

- HTML5;
- CSS3;
- JavaScript;
- PHP v7.2.10 (Ubuntu 18.04.1);
- DBMS MariaDB v10.1.38 (Ubuntu 18.04.1);
- MySql 5.0.12-dev (Ubuntu 18.04.1);
- Web Server Apache/2.4.29 (Ubuntu)

5.2 Strumenti di supporto

Per poter garantire un buon coordinamento all'interno del team di sviluppo è stato utilizzato il servizio *GitHub* per ospitare il repository del progetto, per tenere traccia delle attività da svolgere e per suddividere in modo efficiente e strutturato il lavoro. Inoltre è stato utilizzato *TravisCI*, un servizio di continuous integration, per validare la pagine HTML e i fogli CSS prodotti ad ogni attività di push sul repository remoto. Questo ci ha permesso di controllare costantemente che le modifiche apportate al progetto non avessero introdotto errori che avrebbero potuto rendere non valide le pagine del nostro sito.

5.3 Form

5.3.1 Struttura dei form

Per strutturare i form nel modo migliore e più accessibile, ci siamo informati il più possibile leggendo la documentazione di mozilla, che ha scritto un ottimo articolo intitolato "[How to structure an HTML form](#)". Partendo da quanto abbiamo imparato a lezione e da ciò che è scritto in questo articolo quindi, ogni form con caselle di testo è composto da un `<fieldset>`, ed ogni campo è dentro ad un `<p>`. Inoltre, è stata messa una `<label>` per ogni casella di testo, in modo da migliorare la navigabilità e l'accessibilità.

5.3.2 Validazione dei form

Particolare attenzione è stata posta alla validazione dei dati di tutti i form, che viene eseguita limitando l'input con tag HTML5, con JavaScript e infine con PHP.

5.3.3 Validazione HTML5

Non è facile mettere delle limitazioni sui dati con HTML5, come sappiamo *required* non è abbastanza restrittivo e si accontenta anche di uno spazio. Il primo controllo che abbiamo messo è l'attributo *maxlength*, per far sì che una stringa non possa essere più lunga di un massimo numero di caratteri. Il secondo controllo che abbiamo messo ove possibile, è l'attributo *pattern*, che permette di specificare un'espressione regolare che il testo inserito deve rispettare. E' un buon modo per validare il testo con attributi HTML5, anche se non è sicuro in quanto si può togliere dall'ispezione di pagina.

5.3.4 Validazione JavaScript

Tutti i form con inserimento di dati hanno un sistema di validazione JavaScript. Quando la pagina viene aperta, se c'è un form con un controllo JavaScript, lo script creerà il box di errore in un punto predefinito, pronto ad essere usato. La creazione a runtime evita inconvenienti per cui senza JavaScript questo box poteva vedersi comunque o essere presente inutilmente. Viene poi aggiunto un listener al submit del form, che quindi intercetterà l'invio di dati, annullandolo in caso di errore. I controlli fatti con JavaScript cercano di assicurare che i dati siano corretti: controllando per esempio che l'email sia valida con un'espressione regolare, così come anche per il nome. Inoltre, ci sono anche

dei controlli per essere sicuri che le caselle di testo non contengano testo vuoto o troppo lungo, avvisando l'utente quando un input non è valido.

5.3.5 Validazione PHP

PHP, prima di inserire o aggiornare i dati nel database, fa dei controlli abbastanza semplici, controllando soprattutto che le stringhe non siano vuote o troppo lunghe. In fase di registrazione controlla se l'email è valida utilizzando un'espressione regolare come fa JavaScript. Ove necessario controlla anche che la password e la sua conferma siano identiche.

5.3.6 Scrittura articoli

Gli utenti amministratori hanno la possibilità di scrivere nuovi articoli o di modificare quelli esistenti. Per offrire questa funzionalità abbiamo scelto di utilizzare il tag `<textarea>`. Il motivo di questa scelta deriva da diversi fattori tra cui, la semplicità di gestione e il tempo a disposizione per produrre qualcosa di più elaborato. La seconda strada che si poteva intraprendere era l'inserimento di un plugin *WYSIWYG* (*What You See Is What You Get*), ma abbiamo constatato che uno strumento del genere, oltre ad essere molto difficile da gestire senza JavaScript, poteva generare dei problemi di incompatibilità con i browser e addirittura introdurre degli errori di struttura nei documenti HTML, invalidando così la pagine. Perciò abbiamo implementato la soluzione già descritta in cui l'utente può inserire il contenuto dell'articolo nella textarea utilizzando anche alcuni dei tag HTML che abbiamo concesso di utilizzare, e che verranno interpretati dal browser. Siamo comunque consapevoli che, anche in questo caso l'articolo è prone ad errori, infatti l'utente potrebbe inavvertitamente utilizzare in modo scorretto i tag, rendendo così non valida la pagina dell'articolo.

5.4 Separazione parte privata e pubblica

Molte delle funzionalità di amministratore sono state isolate da quelle riservate al resto degli utenti; in questo modo si aumenta la mantenibilità del sito e soprattutto si conserva una buona struttura logica dell'applicazione. Tuttavia in alcune situazioni abbiamo preferito gestire le funzionalità di entrambe le tipologie di utente nella stessa pagina, gestendo tramite codice quale funzionalità offrire in base al tipo account. In ogni caso le pagine sono tutte strutturate per essere molto corte e leggibili, richiedendo solo la chiamata di metodi per inserire le funzionalità.

- Funzionalità amministratore
 - Scrittura/Modifica articolo (`writeArticle.php`): pagina dedicata, per l'amministratore;
 - Gestione argomenti (`manageArguments.php`): pagina dedicata, per l'amministratore;
- Funzionalità comuni

- Lista dei link agli articoli (`ArticleLinks.php`): condivisa, l'utente visualizza solo la lista dei link mentre l'amministratore visualizza anche un form con cui può inserire un nuovo sottoargomento e funzionalità con cui può creare/gestire articoli relativi ai sottoargomenti;
- Lettura articolo (`ReadArticle.php`): condivisa, l'utente può leggere l'articolo e commentare mentre l'amministratore oltre a fare ciò, può anche eliminare i commenti degli utenti;
- Navbar (`navbar.php`): condivisa, gli utenti visualizzano le voci per navigare nel sito, mentre gli amministratori oltre a queste visualizzano un link che porta al pannello amministratore.

5.5 Strumenti amministratore

Agli utenti di tipo amministratore viene fornito un pannello che permette di accedere ad altre funzionalità oltre a quelle già descritte al punto precedente. Nella pagina `admin-Tools.php`, l'amministratore dispone di tre funzionalità a cui può avere accesso tramite un link:

- **Gestione degli argomenti (`manageArguments.php`)**: in questa pagina è possibile creare un nuovo macroargomento ed eventualmente cancellarne di esistenti;
- **Aggiunta amministratore (`addAdmin.php`)**: in questa pagina è possibile rendere amministratore un utente che ancora non lo è, inserendo il suo indirizzo email. **Nota**: non è possibile rimuovere amministratori poiché, in tal caso, un amministratore potrebbe revocare i permessi a tutti gli altri ed essere quindi l'unico ad avere il controllo totale sul sito. Siamo consapevoli della mancanza di questa opzione ma per questioni di tempo e semplicità non abbiamo implementato un sistema di utenti e permessi più dettagliato;
- **Gestione utenti (`manageUsers.php`)**: in questa pagina è possibile sospendere un utente inserendo il suo nickname ed eventualmente revocare la sospensione agli utenti attualmente sospesi.

5.6 Classi

Per dare una struttura ordinata, in particolare renderla estendibile e mantenibile abbiamo previsto le seguenti classi:

- **Article.php**: fornisce i metodi necessari per la gestione degli articoli (creazione, modifica, stampa del loro contenuto);
- **Card.php**: fornisce i metodi necessari per la stampa delle card presenti nella homepage;
- **Comments.php**: fornisce i metodi necessari per la gestione dei commenti (creazione, eliminazione, stampa);

- **SearchManager.php**: fornisce i metodi necessari per gestire i risultati delle ricerche;
- **Sidebar.php**: fornisce i metodi necessari per stampare la sidebar ed altri elementi associati;
- **Subtopics.php**: fornisce i metodi necessari per la gestione dei sottoargomenti (creazione, eliminazione, stampa);
- **User.php**: fornisce i metodi necessari per gestire gli utenti e le funzionalità ad essi associate;
- **validateData.php**: fornisce metodi di utilità per la validazione di dati.

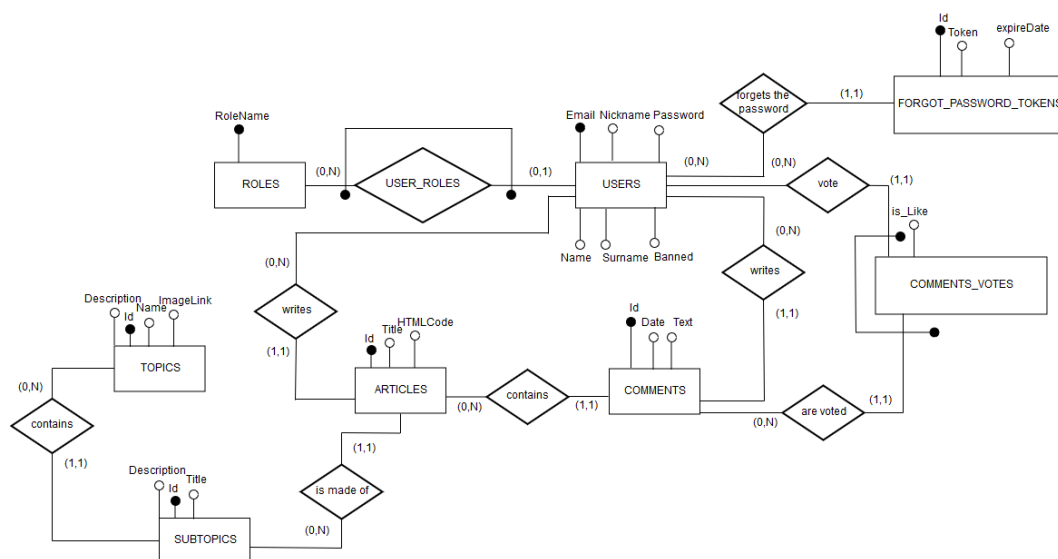
Le classi sopra elencate contengono solo metodi statici e pertanto non necessitano di essere istanziate, questo per rendere semplice il loro utilizzo e facile il loro evolversi. Ci è sembrata la scelta migliore poiché non è stato possibile, per questioni di tempo, implementare un design pattern più complesso (e sicuramente più efficiente e mantenibile) come *MVC*. Le uniche classi istanziabili sono le seguenti:

- **ResultManager**: situata in `ResultManager.php`, si occupa di gestire i messaggi di stato dei metodi;
- **SearchElement**: situata in `SearchManager.php`, si occupa di gestire una ricerca effettuata tramite la `searchbar`;
- **UserInfo**: situata in `User.php`, si occupa di raccogliere le informazioni relative di un utente e viene inoltre utilizzata per salvare un oggetto in una variabile di sessione, utile quando si necessita avere più informazioni contemporaneamente sulla sessione di un utente.

5.7 Gestione degli errori

Come già accennato la classe **ResultManager** viene utilizzata per gestire i messaggi di stato dei metodi. Questa classe è costituita da un campo che identifica la stringa del messaggio e da un campo booleano che stabilisce se questo è un errore oppure no. Molti metodi ne fanno uso per poter ritornare al chiamante un oggetto di questo tipo, al fine di semplificare le modalità in cui vengono stampati i messaggi nel front-end.

5.8 Progettazione del Database



Il database è composto da una struttura abbastanza basilare per conservare tutti i dati nella maniera più semplice possibile e per garantire facile reperibilità di ogni cosa. Le tabelle del database sono:

- In primis, c'è la tabella per gli utenti, chiamata **USERS**, che contiene i dettagli dell'utente come l'email e la password, più un campo che segna se quell'utente è stato bannato;
- La tabella **ROLES** è una tabella adibita a contenere i possibili ruoli che un utente può avere, nel nostro caso solo Amministratore
- Per collegare le due tabelle qui sopra, dalla relazione nasce la tabella **USER_ROLES** che associa ad un utente un ruolo;
- La tabella **TOPICS** è la tabella che contiene tutti i macroargomenti, particolare attenzione qui va al campo *ImageLink*, che conterrà il percorso per la copertina del macroargomento caricata in "uploads/" durante la creazione;
- La tabella **SUBTOPICS** contiene tutti i sottoargomenti con il riferimento al macroargomento di cui fanno parte;
- La tabella **ARTICLES** contiene tutti gli articoli con il riferimento al sottoargomento di cui fanno parte, salvando anche chi è l'autore dell'articolo;
- La tabella **COMMENTS** contiene tutti i commenti, con contenuto data e autore e il riferimento all'articolo di cui ogni commento fa parte;
- Collegata alla tabella **COMMENTS** c'è **COMMENTS_VOTES** che segna quando un utente mette mi piace/non mi piace ad un commento, chiaramente con un riferimento al commento originale, in modo da poter ricavare velocemente tutti i voti di un commento;

- Infine, la tabella **FORGOT_PASSWORD_TOKENS** è utilizzata per memorizzare i token di recupero password, generati quando un utente chiede l'email per recuperare la password dimenticata. Oltre al token è presente una data di scadenza, utilizzata per verificare che il token non sia più utilizzabile dopo 7 giorni. In questa tabella il *token* era *UNIQUE* per garantire che non potessero essercene di doppi, tuttavia la versione di SQL installata nel server di tecweb non permetteva questo poiché il campo era troppo lungo per essere *UNIQUE*. L'assenza di doppi è quindi controllata da PHP alla creazione di un nuovo token.

5.9 Gestione Javascript

Per la gestione delle pagine in assenza di JavaScript sono state utilizzate principalmente due tecniche:

- l'utilizzo del tag `<noscript>` che permette di definire porzioni di codice HTML che verranno interpretate dal browser solamente in caso JavaScript fosse disattivato.
- l'utilizzo di una semplice ma efficace tecnica che permette di definire del codice CSS che verrà interpretato dal browser solamente in caso JavaScript sia attivato. Questa tecnica consiste nel:
 - definire all'interno del file CSS principale solo ed esclusivamente il codice che dovrà essere eseguito in assenza di JavaScript;
 - In un secondo file definire il codice CSS che al contrario verrà eseguito solo ed esclusivamente nel caso in cui JavaScript sia abilitato;
 - Tramite un apposito script JavaScript effettuare l'append di quest'ultimo foglio di stile nel tag `<head>` della pagina web.

L'append ovviamente potrà avvenire solo ed esclusivamente con JavaScript abilitato, e ciò permette di ottenere con facilità il risultato desiderato.

5.10 Ricerca dei contenuti

La ricerca dei contenuti è effettuata tramite la barra di ricerca, presente nella home e nelle pagine con la sidebar. Una volta inserito il termine di ricerca e inviata la richiesta alla pagina `search.php`, quest'ultima utilizzerà la classe la classe **SearchManager** per ottenere i risultati di ricerca da stampare. Questa classe ha tre metodi principali, uno per ogni possibile risultato: macroargomento, sottoargomento e articolo.

Ognuno di questi metodi funziona in maniera abbastanza semplice: legge dal database tutti i possibili elementi e li analizza uno per uno utilizzando la funzione PHP *similar_text(...)* per calcolare la somiglianza tra il termine di ricerca inserito e l'elemento del database analizzato. Quando questa somiglianza è sopra una certa percentuale il risultato viene considerato valido.

Per rendere il calcolo della funzione di somiglianza più efficace, sia il termine di ricerca

che il termine letto da database vengono portati in minuscolo, per non svantaggiare l'utente che potrebbe dimenticarsi di utilizzare le maiuscole. Infine, abbiamo considerato una percentuale di somiglianza abbastanza bassa, precisamente del 55%, questo perché ci siamo accorti che un valore troppo alto filtrava troppo i risultati e secondo noi è più conveniente mostrare qualche risultato in più che qualcuno in meno.

5.11 Menù e sidebar senza JavaScript

Come già accennato il sito si comporta bene senza JavaScript e di seguito verranno illustrati tutti i suoi usi all'interno del sito. JavaScript è stato adoperato per gestire l'apertura e la chiusura degli elenchi presenti nella sidebar e per personalizzare la scrollbar associata ad essa al fine di migliorare l'aspetto grafico del sito. Per la scrollbar ci siamo serviti di una libreria esterna (quindi non sviluppata da noi). Tuttavia nel caso in cui non fosse disponibile, è comunque possibile utilizzare pienamente la sidebar, in quanto la scrollbar viene sostituita con quella di default e gli elenchi vengono aperti tutti automaticamente, al fine di rendere disponibile la consultazione delle voci. Per quanto riguarda i dispositivi mobili, qualora JavaScript non fosse attivato, viene utilizzata una versione del menù a tendina e della sidebar studiate appositamente. Le versioni del menù e della sidebar da utilizzare senza JavaScript vengono stampate in fondo alla pagina e sono accessibili sempre attraverso i consueti pulsanti (ovvero gli stessi utilizzati in presenza di JavaScript) che in questo caso sono delle ancore che portano alla zona della pagina in cui sono situati questi due elementi. Maggiori informazioni riguardanti il funzionamento nello specifico della scrollbar verranno date all'interno della sezione successiva.

5.12 Framework scrollbar

Il team ai fini di rendere la sidebar più piacevole alla vista ha scelto di adottare un framework che permette di implementare una sidebar custom. Il team dopo una approfondita ricerca, ha deciso di scegliere il framework OverlayScrollBar, in quanto era relativamente semplice da utilizzare e forniva una copertura pressoché totale su tutti i browser più comuni, persino sull'ormai datato Internet Explorer fino alla versione 8. Per applicare la scroll bar si è dovuta utilizzare la seguente istruzione definita all'interno delle API del framework:

OverlayScrollbars(element, {})

dove element corrisponde all'elemento su cui si vuole applicare la scroll bar e "{ }" corrisponde ad una lista di possibili parametri che ai fini del progetto si è deciso di lasciare vuota. Si è notato inoltre che applicando la funzione precedentemente descritta direttamente sulla sidebar si presentava un conflitto tra gli stili del framework e quelli da noi descritti. Per risolvere questa problematica si è dovuto inserire un wrapper praticamente privo di stili che contenesse la sidebar ed applicare su di esso la suddetta funzione. Inoltre per far risaltare maggiormente i breadcrumb descritti più nel dettaglio nelle sezioni precedenti si è dovuta utilizzare la seguente istruzione definita all'interno delle API del framework:

instance.scroll(element)

In cui `instance` corrisponde ad un'istanza della scrollbar ritornata dall'istruzione spiegata precedentemente ed `element` corrisponde all' ID dell'elemento a cui si vuole scrollare. Inoltre si sono effettuati molti test sulle performance del sito ed è stato constatato che esse non vengono pressochè intaccate dall'utilizzo di questo framework nonostante coinvolga molto codice javascript. E' importante sottolineare nuovamente che il sito web non sia per niente dipendente da questo framework e che la sua unica funzionalità sia quella di rendere l'estetica della pagina più piacevole. Maggiori informazioni riguardo quest'ultima affermazione sono presenti nella sezione precedente.

6 Compatibilità

6.1 Descrizione

In questa sezione verrà discussa la compatibilità delle tecnologie utilizzate con i browser adottati per testare il sito. Il sito è stato realizzato seguendo in parte le specifiche HTML5, concordando sull'utilizzo di specifici tag relativi a questa versione HTML, soprattutto durante i controlli dei form.

6.2 Tag HTML5 utilizzati

- **<label>**: usato per definire "label", etichette per tutti i tag che la prevedono all'interno dei form. Fornisce un miglioramento dell'usabilità per gli utenti che navigano tramite utilizzo del mouse perché attiva il controllo nel caso un utente clicchi sul testo che prevede una label.
- **<fieldset>**: questo tag è stato utilizzato per raggruppare gli elementi correlati per ciascun form.
- **attributo type per <input>**: abbiamo utilizzato alcuni valori introdotti con HTML5 per dichiarare lo scopo di un elemento `<input>`. Nei form ad esempio, di login, del profile e di registrazione, è presente il `type="email"`. E' stato inoltre utilizzato l'attributo "placeholder" per fornire un suggerimento su ciò che si deve inserire nei campi di input.

6.3 Browser supportati

Il sito deve essere funzionante sul maggior numero di browser disponibili sul mercato con le versioni più recenti. I test sono stati effettuati sui seguenti browser:

- Google Chrome - Versione 72.0.3626.96 (Build ufficiale) (a 64 bit);
- Mozilla Firefox - Versione 65.0 (64 bit);
- Safari - Versione 12.0.3 (14606.4.5);
- Microsoft Edge - Versione 17.17134;

- Internet Explorer - Versione 9 e successive (11.523.17134.0).

Si presuppone che alcune funzionalità non si comportino allo stesso modo in tutti i browser e non siano garantite su quelli più datati. Si accerta comunque che il sito degradi in maniera elegante sui browser più vecchi.

6.4 JavaScript

Dai test effettuati abbiamo riscontrato che il sito è utilizzabile e consultabile completamente senza l'utilizzo di JavaScript secondo le tecniche per ovviare alla sua mancanza descritte nella sezione **Sviluppo**. Alcune limitazioni:

- Non avere la possibilità di chiudere le sezioni degli argomenti nella sidebar poiché vengono aperte automaticamente per poter fruire dei contenuti;
- Tasto "torna su" non funzionante;

Siamo consapevoli di aver utilizzato molto JavaScript ma, a parte la libreria esterna per gestire la scrollbar non sviluppata da noi e utilizzata per lo stile, la maggior parte del codice riguarda ai controlli dei form client-side personalizzati (circa 600 righe sulle 800 che abbiamo). Tutto ciò però non inficia né sull'accessibilità in quanto è stato già descritto come abbiamo gestito la sua assenza, né le prestazioni del sito in termini di caricamento e tempi di risposta come si può notare nella sezione dei test delle performance.

7 Test

7.1 Accessibilità

Strumenti utilizzati per valutare l'accessibilità e risultati ottenuti.

7.1.1 Strumenti utilizzati

- NVDA: screen reader gratuito reperibile al seguente indirizzo: <https://www.nvaccess.org/>;
- WebAIM Color Contrast Checker: strumento per controllare il contrasto dei colori reperibile al seguente indirizzo: <https://webaim.org/resources/contrastchecker/>;
- Web Accessibility Checker: strumento per valutare il grado di accessibilità di un sito secondo le linee guida WCAG 2.0 reperibile al seguente indirizzo: <https://achecker.ca/checker/index.php>;
- Total Validator: strumento per validare le pagine e valutarne l'accessibilità reperibile al seguente indirizzo: <https://www.totalvalidator.com/index.html>;

- Colorblind Web Page Filter: strumento per simulare il modo in cui un utente affetto da disturbi visivi quali Protanopia, Deutanopia, Tritanopia e Achromatopsia visualizza una pagina web, reperibile al seguente indirizzo: <https://www.toptal.com/designers/colorfilter>;
- Google Chrome Audits: strumento integrato nei tools degli sviluppatori che fornisce un numero che indica il grado di accessibilità di un sito web (e anche altri parametri).

7.1.2 Risultati

- NVDA: nonostante la nostra prova con lo screen reader conti molto poco in termini di valutazione (infatti è stato molto difficile da utilizzare), abbiamo notato che alcune accortezze di base (come alt nelle immagini) risultano soddisfatte, e in generale il sito dovrebbe essere consultabile seppur con qualche difficoltà. Siamo comunque consapevoli di non aver potuto avere un giudizio reale sotto questo punto di vista, perciò il sito potrebbe presentare ancora molti da aspetti che devono essere migliorabili per aiutare l'accessibilità a individui che fanno uso di questo strumento;
- WebAIM Color Contrast Checker: in fase di scelta dei colori da utilizzare per foreground e background è stato utilizzato questo strumento al fine di assicurarci che il rapporto dei contrasti fosse almeno pari a 7:1. Pertanto tutti i colori utilizzati sono in linea con WCAG 2.0 AAA;
- Web Accessibility Checker: tutte le pagine sono state sottoposte ai test effettuati da questo strumento (selezionando WCAG 2.0 come riferimento) e non sono stati segnalati errori di alcun tipo. Le uniche segnalazioni sono miglioramenti che possono essere apportati per migliorare ancora di più il grado di accessibilità;
- Total Validator: ehehehe //TODO;
- Colorblind Web Page Filter: il simulatore di daltonismo che permette di simulare il modo in cui gli utenti affetti da Protanopia, Deutanopia, Tritanopia e Achromatopsia, visualizzano il sito è stato utilizzato su tutte le pagine del sito. Pur essendo questo aspetto difficile da valutare se non svolto da individui affetti dalle patologie già citate, il sito ci è sembrato comunque accessibile e usabile in quanto non abbiamo realizzato pagine molto colorate, infatti i colori più utilizzati sono il bianco, il grigio e il blu per i pulsanti. Inoltre i link (in particolare quelli per navigare all'interno del sito) sono tutti sottolineati e quindi ciò aiuta a identificarli anche laddove i colori non riescano;
- Google Chrome Audits: questo strumento, integrato nella toolbar degli sviluppatori di Google Chrome, riporta alcuni parametri principali (Performance, Accessibilità, SEO). Per quanto riguarda l'accessibilità il sito ha riportato un punteggio pari a 100/100. Tuttavia siamo consapevoli che questo strumento non è preciso ed esaustivo come altri già citati ai punti precedenti, come Web Accessibility Checker.

7.2 Usabilità

Strumenti utilizzati per valutare l'usabilità e risultati ottenuti.

7.2.1 Strumenti utilizzati

- Google Mobile Friendliness Test: strumento offerto da Google per testare l'usabilità di un sito su dispositivi mobili, reperibile al seguente indirizzo: <https://search.google.com/test/mobile-friendly>;
- Test umano: il sito è stato utilizzato da persone esterne allo sviluppo per ottenere un parere oggettivo. Inoltre è stato utile per osservare la facilità con cui navigavano e il tempo necessario per capire dove trovare le informazioni.

7.2.2 Risultati

- Google Mobile Friendliness Test: il sito è stato valutato come positivamente, pertanto risulta essere ottimizzato per i dispositivi mobili;
- Test umano: coloro che hanno utilizzato il sito non hanno trovato difficoltà nella navigazione e si sono trovati davanti un sito a detta loro "familiare", questo indica che la struttura scelta è abbastanza affermata e quindi facile da imparare a ricordare.

7.3 Performance

Strumenti utilizzati per valutare le performance e risultati ottenuti.

7.3.1 Strumenti utilizzati

- Google PageSpeed Insights: strumento offerto da Google per testare le performance di un sito web, reperibile al seguente indirizzo: <https://developers.google.com/speed/pagespeed/insights/?hl=it>;
- Google Chrome Audits: strumento integrato nei tools degli sviluppatori che fornisce informazioni sulle performance di un sito web (e anche altri parametri).

7.3.2 Risultati

- Google PageSpeed Insights: i test con questo strumento hanno ottenuto un punteggio ottimo, compreso tra 98 e 100;
- Google Chrome Audits: i test con questo strumento hanno ottenuto un punteggio ottimo, compreso tra 95 e 100.

Nota: i valori di questi test possono variare a seconda di diversi fattori, ad esempio la velocità della connessione.

7.4 Validazione

Strumenti utilizzati per validare i documenti e risultati ottenuti.

7.4.1 Strumenti utilizzati

- Validatore HTML: validatore di documenti HTML offerto dal W3C, reperibile al seguente indirizzo: <https://validator.w3.org/>;
- Validatore HTML&CSS: validatore di documenti HTML compresi di CSS offerto dal W3C, reperibile al seguente indirizzo: <https://jigsaw.w3.org/css-validator/>

7.4.2 Risultati

- Validatore HTML: le pagine del sito sono state tutte validate e non presentano errori;
- Validatore HTML&CSS: i fogli di stile sono stati validati e non presentano errori.

8 Suddivisione dei ruoli

//TODO

- **Giacomo Barzon:**
- **Francesco De Filippis:**
- **Giacomo Greggio:**
- **Michele Roverato:**