

uniøverflow



Gruppo: UniOverflow

indirizzo sito web:

<http://tecnologie-web.studenti.math.unipd.it/tecweb/~erotundo>

credenziali accesso:

docente@studenti.unipd.it
12345678

email referente gruppo:

enrico.rotundo@gmail.com

Giacomo Fornari 1029118
Serena Girardi 1029253
Federico Poli 1032418
Enrico Rotundo 1008052

Progetto Tecnologie Web
aa. 2013-2014

Indice

- [1. Descrizione generale](#)
 - [1.1 Utenza](#)
- [2. Installazione](#)
 - [2.1 Configurare UniOverflow](#)
 - [2.2 Librerie Perl](#)
 - [2.2.1 Template::Toolkit](#)
 - [2.2.2 LibXML](#)
 - [2.2.3 Session](#)
 - [2.3 Controllo](#)
- [3. Progettazione](#)
 - [3.1 Struttura](#)
 - [3.1.1 Struttura pagine](#)
 - [3.1.2 Repository](#)
 - [3.1.3 Definizione](#)
 - [3.1.4 Database](#)
 - [3.2 Presentazione](#)
 - [3.3 Scelte implementative](#)
- [4. Realizzazione](#)
 - [4.1 Gestione input/output](#)
 - [4.2 Sicurezza](#)
- [5. Accessibilità](#)
 - [5.1 Strumenti di navigazione e orientamento](#)
 - [5.1.1 Breadcrumb](#)
 - [5.1.2 Link](#)
 - [5.1.3 Tabindex](#)
 - [5.1.4 Accesskey](#)
 - [5.1.5 Paginazione](#)
 - [5.1.6 Ricerca](#)
 - [5.1.7 Indice manuale](#)
 - [5.2 Equivalenti testuali](#)
 - [5.2.1 Classi aural e auralNotRead](#)
 - [5.3 Tabelle](#)
 - [5.3.1 Summary](#)
 - [5.3.2 Scope e headers](#)
 - [5.3.3 Abbr](#)
 - [5.4 Grafica e testo](#)
 - [5.4.1 Elementi grafici](#)
 - [5.4.2 Lingua](#)
 - [5.4.3 Contrasto testo](#)
 - [5.4.4 Contrasto immagini](#)
 - [5.5 Visibilità motori di ricerca](#)

6. Validazione

6.1 XHTML

6.2 CSS

6.3 XML e XSD

7. Compatibilità

8. Ruoli

1. Descrizione generale

UniOverflow si propone come semplice piattaforma di Question & Answer rivolta agli studenti di informatica dell'Università di Padova. Ci siamo liberamente ispirati al noto Stackoverflow.com come funzionalità di base e layout, implementando però accessibilità per utenti disabili e validazione rispetto allo standard XHTML 1.0 Strict. La sola lettura e ricerca dei contenuti è pubblica. Gli utenti che desiderano inserire domande e poter rispondere devono registrarsi, per tale operazione gli utenti devono essere in possesso di una mail di ateneo con dominio *studenti.unipd.it*. All'inserimento di una domanda essa risulta "aperta", l'autore quando lo riterrà opportuno potrà segnarla come "risolta" determinando la disabilitazione dell'inserimento di ulteriori risposte. L'autore potrà, in un secondo momento, riaprire la domanda permettendo così il proseguimento della discussione. Il sito viene consegnato con una base di contenuti generici precaricati, copiati da un forum di informatica italiano. Il testo inserito dagli utenti può essere formattato tramite l'utilizzo di caratteri di markup, come spiegato nella pagina di Manuale.

1.1 Utenza

L'utenza di UniOverflow è composta principalmente da studenti universitari dei corsi di laurea in informatica, tuttavia il sito deve essere ugualmente accessibile a chiunque abbia interessi rivolti all'informatica.

2. Installazione

Di seguito sono riportati i passi necessari per installare il progetto UniOverflow. La guida è rivolta a chi utilizza il sistema operativo Ubuntu 13.10 e ha già installato il server Apache2 con il modulo per l'integrazione con Perl. Con lievi variazioni la guida può valere per un qualsiasi sistema operativo Linux moderno.

2.1 Configurare UniOverflow

Copiare il file *contrib/unioverflow.conf* nella cartella delle configurazioni di Apache, che di solito è */etc/apache2/conf.d*, se non c'è già crearla con *sudo mkdir /etc/apache2/conf.d*.

Il comando da eseguire sarà del tipo:

```
sudo cp contrib/unioverflow.conf /etc/apache2/conf.d/unioverflow.conf
```

Bisogna configurare le prime due righe della configurazione mettendo dove c'è il comando *Alias* il percorso assoluto della repository del progetto:

```
sudo gedit /etc/apache2/conf.d/unioverflow.conf
```

(Ubuntu 13.10 e successivi) se accedendo a *localhost/unioverflow* da "forbidden" bisogna aggiungere a entrambi i tag *<Directory>* la stringa:

Require all granted

Poi dovete riavviare il server, con il comando:

```
sudo service apache2 restart
```

(Ubuntu 13.10) se il browser visualizza il testo dei cgi e non li esegue:

```
sudo a2enmod cgi
```

2.2 Librerie Perl

2.2.1 Template::Toolkit

È un sistema di template utilizzabile con Perl. Per installarlo eseguire da terminale il comando:

```
curl -L http://cpanmin.us | perl - --sudo Template
```

2.2.2 LibXML

Per installarlo eseguire da un terminale il comando:

```
curl -L http://cpanmin.us | perl - --sudo XML::LibXML
```

o in alternativa (Ubuntu):

```
sudo apt-get install libxml-libxml-perl
```

2.2.3 Session

(Ubuntu)

```
curl -L http://cpanmin.us | perl - --sudo CGI  
curl -L http://cpanmin.us | perl - --sudo CGI::Session
```

2.3 Controllo

Inizialmente il repository viene fornito con un database di esempio *db/db.xml.original* ma senza la controparte che il sistema effettivamente utilizza:

```
cp db/db.xml.original db/db.xml
```

Conviene utilizzare il comando di Makefile per settare correttamente i permessi:

```
make flags-local
```

Se la configurazione di Uni Overflow è stata fatta bene la URL <http://localhost/unioverflow/> presenta l'homepage del progetto UniOverflow.

3. Progettazione

UniOverflow si rivolge ad utenti italiani, quindi possiede le caratteristiche di un sito localizzato nel contesto italiano: la lingua utilizzata è l'italiano, la direzione di lettura va da sinistra verso destra. Le date utilizzano il formato YYYY-MM-DD, questo formato è stato scelto per la facilità con cui è possibile ordinare liste di date, è stato quindi preferito al formato italiano. Riteniamo che tale scelta non disorienta gli utenti target poiché si suppone che abbiano familiarità con siti web e contenuti di lingua anglosassone, comunemente legati al mondo dell'informatica.

3.1 Struttura

3.1.1 Struttura pagine

Ogni pagina presenta:

- logo;
- menù di navigazione pagine, Accedi e Registrati (utente non autenticato) o Esci (utente autenticato);
- percorso all'interno del sito (es. Ti trovi in: Domande » Vedi domanda #8);
- {contenuto della pagina};
- footer (con ripetizione dei link ad about e manuale).

3.1.2 Repository

Il repository è strutturato con le seguenti cartelle:

- **cgi-bin**: contiene script Perl che richiamano, per ogni pagina, il relativo Controller;
- **contrib**: contiene il file di configurazione di Apache *unioverflow.conf* per Linux, *http.conf* per Mac OS X 10.9;
- **Controller**: contiene i Controller, uno per ogni pagina disponibile;
- **db**: contiene il database XML ed il relativo schema XSD;
- **Lib**: contiene varie classi di utility come XMLCRUD e Utils;
- **Middleware**: contiene le classi che implementano cookie, sessione e autenticazione;
- **Model**: contiene la classi che implementano il model del database;
- **public_html**: contiene le pagine statiche, o parti di esse, gli stili CSS e le immagini;
- **View**: contiene le viste delle pagine da renderizzare con Template::Toolkit.

Largo impegno è stato dedicato alla costruzione del back-end, cercando di limitare l'utilizzo di librerie Perl ed evitando di utilizzare i framework web disponibili sulla rete (come per esempio Dancer, liberamente scaricabile da <http://www.perldancer.org/>). Questa scelta ha inciso non poco sulle tempistiche di realizzazione del sito, aumentando sicuramente il carico di lavoro, ma così facendo abbiamo potuto apprendere meglio molti aspetti del linguaggio Perl che prima ci erano ignoti (utilizzo di moduli e namespace, distinzione tra “use” e “require”, corretta gestione del charset, passaggio di funzioni per indirizzo, programmazione orientata agli oggetti con ereditarietà in Perl).

3.1.3 Definizione

Sono disponibile le seguenti pagine, * contrassegna le pagine che forniscono funzionalità per l'utente autenticato:

- **registrati**: mostra il form di registrazione con le relative istruzioni;
- **accedi**: mostra il form per effettuare il login;
- **index**: mostra la lista delle domande presenti nel database, ordinate a partire dall'ultima inserita nel sistema. Inoltre, tramite il form di ricerca, è possibile effettuare ricerche di stringhe tra i titoli delle domande già presenti;
- **manuale**: mostra un elenco indicizzato delle istruzioni relative alle funzionalità di UniOverflow;
- **vedi-domanda***: mostra interamente la domanda selezionata con le relative risposte paginate, l'utente autenticato può, se la domanda è "aperta", inserire risposte;
- **scrivi-domanda***: se non autenticato mostra i link per accedere o creare un account, per l'utente autenticato mostra il form di inserimento domanda, con riferimento alle istruzioni su come formattare il testo;
- **esci***: effettua il logout, disponibile solo per l'utente autenticato.

3.1.4 Database

Il database è situato nella cartella */db* e si compone di un solo file XML di tipo "tende alla veneziana", valido secondo lo l'omonimo schema XSD. XML Schema è stato scelto per la maggiore espressività, chiarezza ed estendibilità. Inoltre, sono state utilizzate funzionalità specifiche di XML Schema, come la definizione di specifici tipi per gli elementi.

La struttura del database con gli elementi *questions* e *answers* allo stesso livello è stata scelta per mantenere una separazione tra domande e risposte.

3.2 Presentazione

Lo schema organizzativo scelto è di tipo esatto e cronologico. Le domande e le risposte sono indicizzate cronologicamente per l'utente il quale trova i contenuti ordinati temporalmente, ancorando nuovi contenuti alla propria mappa mentale con una maggiore semplicità. In molti casi, l'utente sa già quello che sta cercando recandosi in questo sito web, pertanto, uno schema organizzativo esatto è adeguato. Qualche elemento di schema orientato al task si trova nell'inserimento di una domanda o di una risposta, nella registrazione o nell'accesso.

La struttura organizzativa è prevalentemente gerarchica. Il massimo livello di annidamento lo troviamo nella visualizzazione della discussione relativa ad una domanda, nella quale bisogna accedervi attraverso la lista delle domande. Sono presenti tracce di ipertestualità quando si trovano i link al manuale.

Per il sito web sono stati scritti due diversi layout fluidi basati sulla dimensione dello schermo e dal tipo di dispositivo che intende accedere alle pagine:

- per dispositivi **desktop**, ovvero con larghezza dello schermo maggiore di 480 pixel;
- per dispositivi **palmari**, ovvero con larghezza dello schermo minore di 480 pixel.

3.3 Scelte implementative

La registrazione è supposta per i soli utenti con mail *@studenti.unipd.it*, purtroppo il server messo a disposizione dal laboratorio non permette l'invio di email, pertanto ci è impossibile verificare l'autenticità della mail inserita.

4. Realizzazione

4.1 Gestione input/output

L'immagine che segue illustra come viene gestita la codifica e decodifica del testo inserito dagli utenti.

Il browser invia testo UTF-8 puro, a livello client non vengono eseguite operazioni sugli input per esempio tramite Javascript. La classe Sanitize possiede dei metodi che effettuano gli opportuni replace dei caratteri non consentiti. C'è un metodo per ogni tipologia di campo testuale di input.

Il controller effettua i controlli come consistenza ed esistenza dei campi, in caso di anomalie provvede a generare l'errore opportuno, che sarà visualizzato in seguito dalla vista.

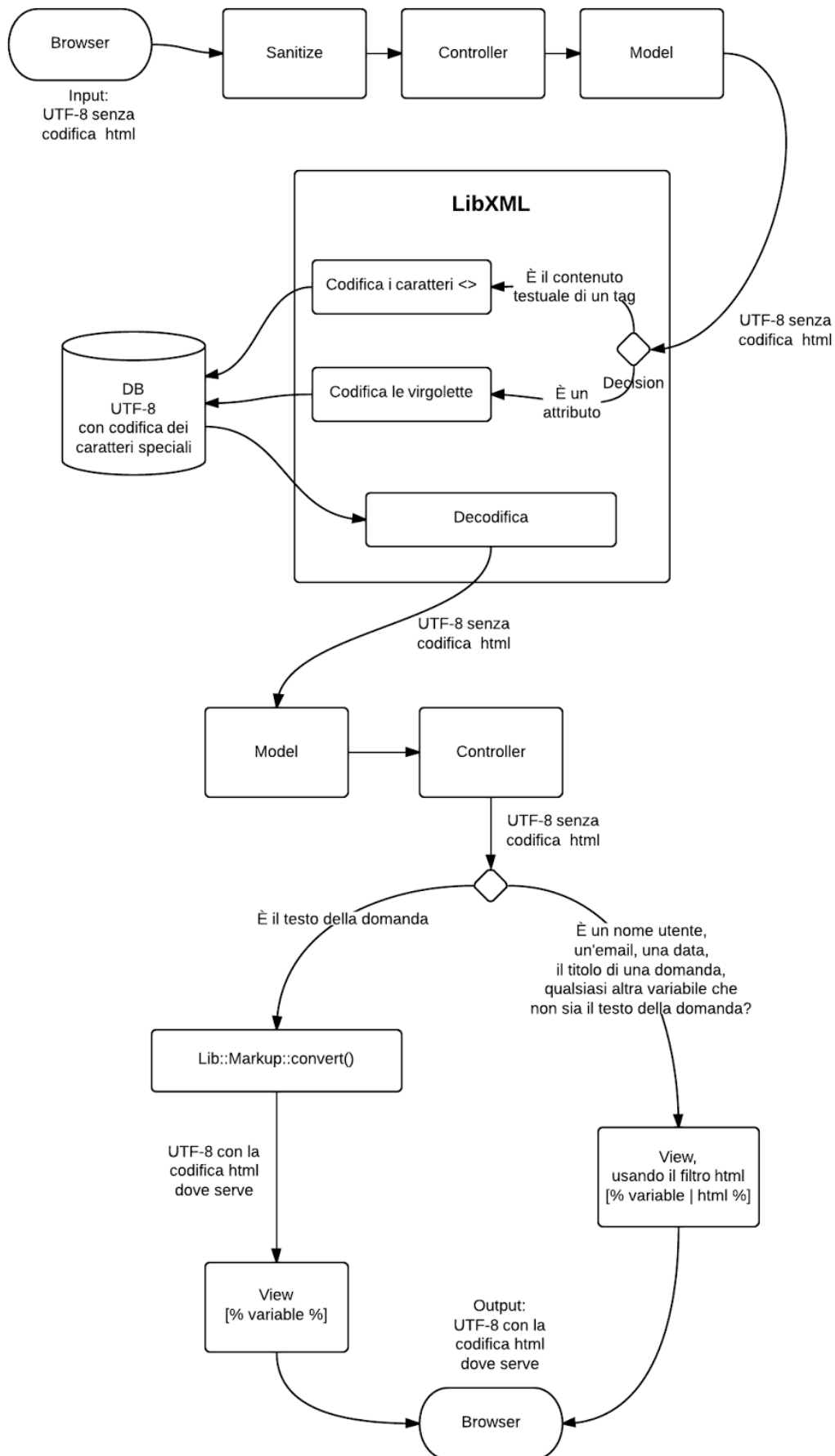
Il model assume la validità dei dati da inserire nel database, viene chiamata la libreria LibXML che si interfaccia direttamente con file xml prescelto.

LibXML si occupa di codificare opportunamente i caratteri speciali con i codici compatibili con xml.

La fase di decodifica inizia leggendo il contenuto del file xml usato come db.

Il Model, tramite i metodi Get recupera le informazioni dal db e le passa al controller che identifica la tipologia del dato. Il testo delle domande va convertito in html laddove siano presenti indicatori di markup, a tal fine il metodo Lib::Markup::convert() esegue la conversione. In questo caso, la vista renderizza il testo senza operare trasformazioni.

Tutto ciò che non è testo di una domanda viene indirizzato alla vista che tramite il comando [% variabile | html %] di Template::Toolkit si occupa di convertire in html tutte le occorrenze dei caratteri speciali.



4.2 Sicurezza

La sicurezza di UniOverflow viene in seguito valutata rispetto alla [OWASP Top Ten Project 2013](#) che rappresenta un punto di riferimento sulla sicurezza delle applicazioni web. La struttura seguente rispecchia quella del documento pdf [OWASP Top Ten 2013](#).

A1. Injection: questo exploit sfrutta l'inefficienza dei controlli sui dati ricevuti in input ed inserisce codice maligno ad esempio all'interno di una query XPath. Pertanto è necessario controllare che i form di inserimento non presentino vulnerabilità in questo senso. Nella tabella di seguito, le colonne rappresentano il form ed il relativo URL parziale (per comodità "cgi-bin/" non viene riportato negli URL in tabella). Le righe rappresentano l'injection testata nella relativa form.

	Cerca-index.cgi	Registrazione - registrati.cgi	Accedi - accedi.cgi	Chiedi - scrivi-domanda.cgi	Rispondi - vedi-domanda.cgi
Controllo	✓	✓	✓	✓	✓

A2. Broken Authentication and Session Management:

- 1. Le credenziali di autenticazione non vengono salvate in forma criptata nel database. Sebbene totalmente insicura, questa scelta ci ha permesso un debug e testing dell'applicazione in modo semplificato, soprattutto per la parte relativa alla registrazione utente. Se il progetto dovesse passare da un ambito accademico ad un ambito di produzione sarebbe comunque facile utilizzare tecniche di hash.
- 2. Il recupero delle credenziali non è stato implementato pertanto non una volta registrati non è più possibile cambiare e/o visualizzare la password.
- 3. il Session ID non viene mai trasmesso nell'URL.
- 4. la Session ID ha una scadenza fissata a 3 ore.
- 5. UniOverflow trasmette le credenziali di autenticazione e di sessione attraverso una connessione non sicura, questa è una scelta obbligata in quanto il server del laboratorio non lo permette.

A3. Cross-Site Scripting (XSS): gli input utente dal browser vengono opportunamente filtrati con caratteri di escape.

A4. Insecure Direct Object References: poiché UniOverflow non prevede la distinzione tra contenuti privati e pubblici non vi è la possibilità di raggiungere risorse senza autorizzazione. L'unica risorsa parametrizzata infatti è *unioverflow/cgi-bin/vedi-domanda.cgi?id=xxx* ma non è privata.

A5. Security Misconfiguration: la configurazione del server web è a carico dell'amministratore del laboratorio.

- A6. **Sensitive Data Exposure:** Sotto questo aspetto UniOverflow non presenta particolari difese: gli unici dati sensibili presenti in UniOverflow sono le password utenti memorizzate nel database in *db/db.xml*; inoltre i dati vengono trasmessi tramite connessioni non crittate. Questo tipo di vulnerabilità non sono state contrastate poiché non rientrano nei requisiti del progetto didattico.
- A7. **Missing Function Level Access Control:** sono stati implementati dei controlli per ignorare richieste inviate da utenti malintenzionati. In particolare i controller relativi a *unioverflow/cgi-bin/vedi-domanda.cgi* e *unioverflow/cgi-bin/scrivi-domanda.cgi* controllano la corretta autenticazione dell'utente prima di effettuare qualsiasi operazione di inserimento di contenuti. In una prima versione di UniOverflow era stata rilevato un mancato controllo sull'autenticazione, era infatti possibile inserire contenuti tramite script da console senza essere in possesso di credenziali valide, tale circostanza era riproducibile tramite il comando di esempio
`curl -data "titolo=INJECT_TITLE&testo=INJECTED_TEXT" http://localhost/unioverflow/cgi-bin/scrivi-domanda.cgi`
- A8. **Cross-Site Request Forgery (CSRF):** questo scenario non è stato adeguatamente contrastato poiché non rientra nello scopo del progetto didattico. Tuttavia per raggiungere un buon livello di protezione basterebbe implementare un sistema di generazione di token collegati ad una singola richiesta eseguibile (es. captcha e/o token in un campo *hidden* del form).
- A9. **Using Components with Known Vulnerabilities:** le librerie utilizzate sono quelle riportate nel sito del docente <http://docenti.math.unipd.it/gaggi/tecweb/librerie.html>. L'aggiornamento e la sicurezza sono a carico dell'amministratore del laboratorio.
- A10. **Unvalidated Redirects and Forwards:** gli unici redirect avvengono tramite header http dalle Views agli script cgi-bin. Non ci sono redirect via query string che richiederebbero dei controlli in un apposita whitelist.

La scelta di [OWASP Top Ten 2013](#) rispetto alle valutazioni operate in materia di sicurezza deriva dalla ricerca di una fonte completa ed autorevole sulla quale valutare UniOverflow. La completa copertura di tutti i punti citati non rappresenta uno degli obiettivi del gruppo e del progetto didattico. Consideriamo le precedenti valutazioni una buona base sulla quale affermare che UniOverflow è sufficientemente sicuro da attacchi malevoli e più in generale da inserimenti di dati che accidentalmente potrebbero comprometterne funzionalità.

5. Accessibilità

Per cercare di massimizzare l'accessibilità e l'utilizzabilità anche agli utenti non vedenti che utilizzano screen-reader, abbiamo introdotto elementi che permettono di muoversi nella pagina più facilmente e di creare una mappa mentale del sito con maggiore semplicità. Inoltre, abbiamo prestato attenzione con altre disabilità visive quali i daltonici.

5.1 Strumenti di navigazione e orientamento

5.1.1 Breadcrumb

Al fine di orientare l'utente nella navigazione, è stato inserito il percorso aggiornato in base alla posizione corrente tra le pagine. In particolare, nelle pagine che prevedono una paginazione, viene indicato anche il numero della pagina corrente. Tale strumento di orientamento è posto nell'area sicura della pagina per molti dispositivi, poiché visibile senza necessità di scroll.

Per gli utilizzatori di screen-reader, il contenuto viene fornito più leggibile grazie a dei contenuti nascosti. Ad esempio, la lettura di ciò che viene visualizzato come *"Ti trovi in: Domande » Vedi domanda #8, pagina 3"* verrà letto come *"Ti trovi in: Domande » Vedi domanda numero 8, pagina 3"*.

5.1.2 Link

Per evitare disorientamento da parte dell'utente, i colori dei link sono stati lasciati invariati, concedendo al browser di applicare le impostazioni di default. In tal modo, i link avranno un aspetto familiare all'utente che saprà riconoscere immediatamente il loro stato.

Al fine di migliorare la navigazione, sono stati introdotti dei link sia **visibili** che **nascosti**, utili agli screen-reader.

- **passa direttamente al contenuto:** all'inizio delle pagine è presente un link *nascosto* che permette di agli utilizzatori di screen-reader di non leggere il menù e di saltare direttamente al contenuto della pagina;
- **torna su:** alla fine delle pagine è presente un link *visibile* che permette di tornare direttamente all'inizio del contenuto della pagina. Tale link è presente anche alla fine di ogni sezione del manuale e alla fine di ogni risposta poiché, in quest'ultimo caso, i contenuti posso essere arbitrariamente lunghi;
- **passa alla risposta successiva:** dopo la lettura delle informazioni di data e autore di una risposta, è presente un link *nascosto* che permette di passare alla risposta successiva. Nel caso la risposta sia l'ultima della pagina, ma non l'ultima associata alla domanda, il link rimanda alla prima risposta della pagina successiva.

5.1.3 Tabindex

L'attributo *tabindex* è stato utilizzato per migliorare ulteriormente la navigazione nelle pagine permettendo di fare meno uso del mouse. In particolare, è utilizzato nel menù, nelle liste per scorrere le domande e le risposte (ed i relativi link sopracitati), i link del manuale e i form presenti nel sito web, e infine negli strumenti per la paginazione.

5.1.4 Accesskey

L'attributo *accesskey* è stato utilizzato per le stesse motivazioni del *tabindex* cercando di assegnare una chiave il più possibile intuitiva. Nel menù le *accesskey* sono composte da numeri dall'1 al 5. Nella pagina principale è stata assegnata una *accesskey* al pulsante di ricerca. Inoltre, è presente una *accesskey* in tutti i form del sito web.

5.1.5 Paginazione

Dato il numero potenzialmente alto di domande e di risposte, è stato inserito un strumento di paginazione per poter frammentare la visualizzazione dei contenuti ed evitare che lo scroll e il caricamento della pagina diventassero troppo pesanti nel caso di dispositivi con connessioni lente o con dispositivi dallo schermo di dimensioni contenute.

La paginazione, se necessario, presenta:

- il link alla prima pagina;
- i link alle 2 pagine precedenti a quella corrente;
- i link alle 2 pagine successive a quella corrente;
- il link all'ultima pagina.

Nel caso di compatibilità sufficiente (vedi [Compatibilità](#)) e di utenti vedenti, la comprensione della successione delle pagine è aiutata dalla colorazione dello sfondo dei link.

Nel caso di accesso da browser testuale o da screen-reader, il contenuto è così presentato (ad esempio, con pagina corrente 7 e 15 pagine totali):

- il link a pagina 1 è *"Inizio"*;
- il link a pagina 5 è *"Pagina 5"*;
- il link a pagina 6 è *"Precedente: Pagina 6"*;
- l'indicazione della pagina corrente è *"Attualmente stai leggendo pagina 7"*;
- il link a pagina 8 è *"Successiva: Pagina 8"*;
- il link a pagina 9 è *"Pagina 9"*;
- il link a pagina 15 è *"Fine"*.

5.1.6 Ricerca

Per un sito web che offre un servizio di Q&A è importante fornire uno strumento di ricerca. La fruizione dei contenuti è ordinata, ma potenzialmente lunga. Per tale motivo, è presente una casella di ricerca che aiuta l'utente a trovare ciò che sta cercando. La ricerca viene effettuata sul titolo della domanda. Quando si esegue una ricerca, le breadcrumb vengono aggiornate per indicare all'utente che sta visualizzando il risultato di una ricerca.

5.1.7 Indice manuale

Nel manuale è presente una lista delle sezioni di cui è composto il manuale. L'utente può servirsene per saltare direttamente alla sezione di interesse. Questo strumento risulta utile soprattutto agli utenti che utilizzano uno screen-reader poiché gli permette di leggere solo il contenuto a loro necessario.

5.2 Equivalenti testuali

5.2.1 Classi aural e auralNotRead

Al fine di agevolare l'accessibilità agli utenti utilizzatori di screen-reader, sono stati nascosti elementi come immagini e parti di testo, per sostituirli in alcuni casi con testo maggiormente comprensibile. Per fare ciò, si sono introdotte due classi:

- **aural**, per nascondere elementi utili agli screen-reader;
- **auraNotRead**, per nascondere elementi agli screen-reader.

In particolare, al logo “UniOverflow” nell'intestazione delle pagine è assegnata la classe *auralNotRead*, mentre all'intestazione (<h1>UniOveflow</h1>) è assegnata la classe *aural*.

5.3 Tabelle

5.3.1 Summary

L'attributo *summary* viene utilizzato per dare una breve descrizione del contenuto della tabella. Tale informazione può essere utilizzata dagli screen-reader per dare un'idea di cosa presenta la tabella.

5.3.2 Scope e headers

Gli attributi *scope* e *headers* possono essere utilizzati dagli screen-readers per mettere in relazione le intestazioni delle celle ed il loro contenuto. In particolare, lo scope utilizzato nella tabella presente nel manuale si riferisce alle colonne, ovvero quella cella viene associata come intestazione della colonna.

5.3.3 Abbr

L'attributo *abbr* può essere usato dagli screen-reader per evitare di ripetere l'intera intestazione ad ogni cella. Nonostante la tabella presente nel manuale non sia lunga, l'inserimento di una abbreviazione mantiene una migliore accessibilità anche nel caso si aggiungano contenuti alla tabella.

5.4 Grafica e testo

5.4.1 Elementi grafici

Si è scelto di inserire pochi elementi grafici nelle pagine, privilegiando una grafica semplice. L'intento è quello di evitare un sovraccarico cognitivo e facilitare l'utente nella consultazione dei contenuti. Un altro elemento che favorisce il raggiungimento di tale obiettivo è il posizionamento dei contenuti che si presentano in modo lineare ed intuitivo, per i quali non si spende molto tempo a capire la struttura e l'ordine di esibizione.

5.4.2 Lingua

La lingua base del sito web è l'italiano, ma in qualche pagina viene fatto uso di vocaboli in inglese. Per tali termini è stato utilizzato l'attributo *xml:lang* per indicare agli screen-reader la lingua della parola.

Peculiarità del sito web è la possibilità di inserire contenuti da parte degli utenti che possono essere in una lingua differente da quella italiana. Nelle scelte attuate per questa implementazione non è previsto un riconoscimento dei testi in altre lingue pertanto tutto viene letto in lingua italiana, ma sarebbe possibile effettuare delle modifiche più o meno complesse, quali:

- dare la possibilità all'utente che inserisce il contenuto di marcare il testo con la lingua appropriata;
- effettuare un parsing del testo per il riconoscimento della lingua ed associare le informazioni necessarie.

5.4.3 Contrasto testo

Al fine di verificare il rapporto di contrasto tra i colori è stato utilizzato Colour Contrast Analyser, uno strumento per il controllo delle combinazioni di colori di primo piano e sfondo. Per ogni elemento analizzato viene riportato il valore del rapporto tra la differenza di colore e la differenza di luminosità, ed il rapporto di contrasto luminoso, nei casi di normalità visiva e di daltonismo (Protanopia, Deuteranopia, Tritanopia). Si sono fissati dei valori minimi di accettazione, quali:

- Colore/Luminosità: 500/125
- Luminosità:
 - WCAG 2.0 AA: 4.5:1
 - WCAG 2.0 AAA: 7:1

Seguono i risultati degli elementi sui quali si sono posti più dubbi.

Elemento: testo bianco del menù su sfondo rosso scuro.

Colori in esadecimale: #77261A - #FFFFFF.

Daltonismo	Colore/Luminosità	Luminosità
Nessuno	582/195	10.1:1
Protanopia	639/208	13.3:1
Deuteranopia	614/197	11.1:1
Tritanopia	561/193	10.0:1

Elemento: testo nero delle breadcrumb su sfondo grigio.

Colori in esadecimale: #000000 - #DCDCDC.

Daltonismo	Colore/Luminosità	Luminosità
Nessuno	660/220	15.3:1
Protanopia	658/219	15.2:1
Deuteranopia	659/219	15.2:1
Tritanopia	658/219	15.3:1

Tutti i valori riportati rientrano nei valori di accettazione.

5.4.4 Contrasto immagini

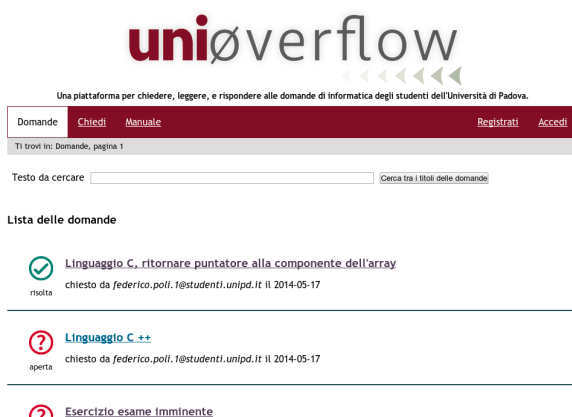
Al fine di verificare come vengono viste le pagine web da un utente affetto da daltonismo, è stato utilizzata l'estensione SEE di Google Chrome. Seguono le immagini delle simulazioni della parte di pagina web più significativa per tale test.



Protanomia



Deuteranomalia



Tritanomalia



Acromatopsia

Per tutti i tipi di daltonismo, l'utente riesce a comprendere il significato del contenuto. In presenza di acromatopsia, deuteranomalia o protanomalia, non è chiara la differenza di colore tra il simbolo della domanda risolta e quello della domanda aperta. Ciò non risulta un problema poiché il significato del simbolo è chiaramente comprensibile anche senza colore, ed inoltre è rimarcato dalla spiegazione testuale.

5.5 Visibilità motori di ricerca

Al fine di rendere aumentare l'accessibilità anche da parte dei motori di ricerca, si è fatto uso del metatag *robot*. Tale attributo comunica allo spider quali pagine si vuole che vengano indicizzate e quali invece si vuole che vengano saltate.

Si è scelto di indicizzare le pagine il cui contenuto è rilevante per l'utente, in cui cambia frequentemente e in cui l'utente può accedere più frequentemente. Le pagine identificate con tali caratteristiche sono:

- la pagina della *lista delle domande*;
- la pagine di *visualizzazione della domanda*;
- la pagine del *manuale*.

6. Validazione

6.1 XHTML

Le pagine sono state validate tramite l'uso di Total Validator (<http://www.totalvalidator.com/>).

L'XHTML è stato validato secondo XHTML 1.0 Strict e tutte le pagine validano correttamente, per maggior sicurezza è stato utilizzando anche il validatore W3C (<http://validator.w3.org/>) ottenendo lo stesso risultato.

Abbiamo inoltre validato l'accessibilità secondo lo standard WCAG 2.0 AAA e tutte le pagine validano correttamente.

6.2 CSS

Il CSS è stato validato utilizzando il validatore CSS fornito dal W3C rispetto alla specifica CSS versione 3 (<http://jigsaw.w3.org/css-validator/>) e tutti i fogli di stile sono risultati validi.

6.3 XML e XSD

Per la validazione di */db/db.xml* è stato utilizzato un file XML uguale al succitato file di esempio */db/db.xml.original*, la validazione è stata effettuata da:

- terminale, con il comando `xmllint --noout db/db.xml`
- online, su http://www.w3schools.com/xml/xml_validator.asp

La validazione di */db/db.xsd* è stata effettuata da:

- terminale, con il comando `xmllint --noout db/db.xsd`
- online, su http://www.w3schools.com/xml/xml_validator.asp

La validazione del database XML sul relativo schema XSD è stata effettuata da:

- terminale, con il comando `xmllint --noout --schema db/db.xsd db/db.xml`
- online, su <http://www.freeformatter.com/xml-validator-xsd.html>

Tutte le validazioni hanno avuto risultato positivo.

7. Compatibilità

Di seguito viene riportato un riepilogo sui diversi browser su cui è stata testata la compatibilità nella visualizzazione del sito e sugli eventuali errori riscontrati per ognuno, divisi per sistema operativo:

Mac OS X Mavericks

✓ Safari 7

Ubuntu 13.10 LTS

✓ Chrome 33

✓ Firefox 28

Windows XP

✓ Firefox 19

✓ Chrome 28

✓ Opera 19

IE6 :

- La media query relativa agli schermi di grandezza più piccola non è supportata se si disabilita javascript.
- Il menù presenta problemi nel colore e nella visualizzazione,
- le immagini non si vedono correttamente,
- il breadcrumb viene posizionato scorrettamente nella pagina,
- nella pagina vedi-domanda l'immagine relativa ad aperta/chiusa viene posizionata in maniera errata,
- l'input "text" di cerca risulta piccolo rispetto al width dato da css,
- ▲ in torna su non viene visualizzato.

IE7 :

- La media query relativa agli schermi di grandezza più piccola non è supportata se si disabilita javascript.
- L'hover degli elementi laterali del navigation, "registrati" e "accedi", non funziona correttamente,
- il pulsante "cerca tra i titoli delle domande" risulta spostato.
- ▲ in torna su non viene visualizzato.

IE8



Una piattaforma per chiedere, leggere, e rispondere alle domande di informatica degli studenti dell'Università di Padova.

Domande	Chiedi	Manuale	Ti trovi in: Domande, pagina 1	Registrati	Accedi
Testo da cercare		Cerca tra i titoli delle domande			

Lista delle domande



aperta

[Linguaggio C, ritornare puntatore alla componente dell'array](#)

chiesto da federico.poli.1@studenti.unipd.it il 2014-05-17

- La media query relativa agli schermi di grandezza più piccola non è supportata se si disabilita javascript.

Windows 7

- ✓ IE9
- ✓ IE10
- ✓ IE11

Windows 8

- ✓ IE10 Desktop

Windows 8.1

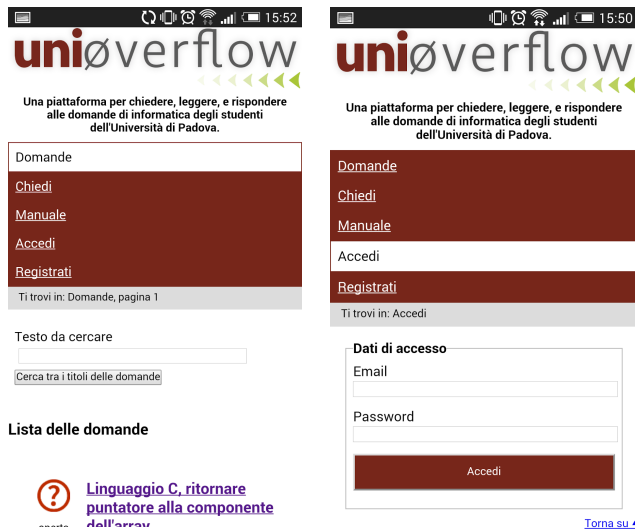
- ✓ IE11 Desktop

Abbiamo ritenuto di non correggere gli errori per i browser IE6 e IE7 in quanto browser obsoleti, tenendo soprattutto conto del tipo di utenza che ci si aspetta: studenti e informatici, dai quali ci si può aspettare l'utilizzo di browser aggiornati.

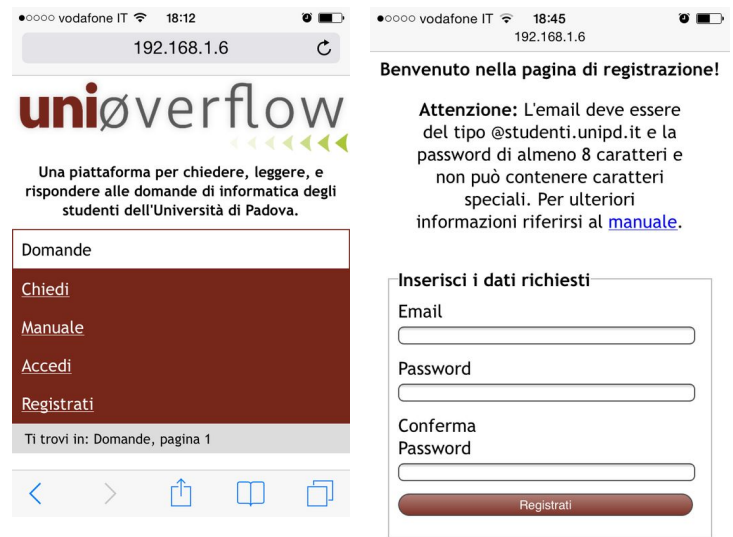
Le versioni di Internet Explorer precedenti alla 9, non supportano le media query, per questo è stato utilizzato lo script <http://code.google.com/p/css3-mediaqueries-js/> che permette di utilizzare le media query come se il browser le supportasse nativamente, occupandosi di fare il parsing del CSS e di applicarlo. Essendo uno script, se si disabilita javascript la media query relativa agli schermi di grandezza più piccola non funziona.

Android

Chrome 35



iOS 7



Modalità testuale

Abbiamo provato ad utilizzare il sito da terminale(modalità testuale) utilizzando Links (<http://links.twibright.com/>) per riscontrarne la facilità d'uso e navigazione, appurando che grazie alla costruzione delle pagine utilizzando liste e titoli appropriati, il contenuto è facilmente raggiungibile ed è risultato semplice spostarsi velocemente da una posizione all'altra grazie ai link di posizionamento.

```
Logo di UniOverflow Passa direttamente al contenuto
UniOverflow
Una piattaforma per chiedere, leggere, e rispondere alle domande di informatica degli studenti dell'Universit'a di Padova.
Navigazione
* Domande
* Chiedi
* Manuale
* Accedi
* Registrati
Ti trovi in: Domande, pagina 1
Cerca tra i titoli delle domande Testo da cercare [ Cerca tra i titoli delle domande ]
Lista delle domande
* Domanda aperta
Linguaggio C, ritornare puntatore alla componente dell'array
chiesto da federico.poli.1@studenti.unipd.it il 2014-05-17
* Domanda aperta
Linguaggio C ++
chiesto da federico.poli.1@studenti.unipd.it il 2014-05-17
* Domanda aperta
Esercizio esame imminente
chiesto da serena.girardi@studenti.unipd.it il 2014-05-17
Passa alla risposta numero 6, pagina 1
L'esercizio dice che la funzione RICORSIVA deve essere questa
float* minimo(const float* array, int size)
con gli ARGOMENTI indicati e NON che deve essere una funzione ausiliaria. Sei fuori strada.
Torna su
* il 2014-05-30 enrico.rotundo@studenti.unipd.it ha scritto:
Passa alla risposta numero 7, pagina 1
No, mi spiace ma questa volta sbagli tu. C'e scritto chiaramente che si posso utilizzare funzioni ausiliarie. Minimo e la funzione che
dovrebbe essere ricorsiva ma che non e, e controllo e quella ausiliaria. Anche il prof ha detto che va bene in questo modo. A me serve
sapere come tornare il puntatore a quella componente, grazie
Torna su
* il 2014-05-30 enrico.rotundo@studenti.unipd.it ha scritto:
Passa alla risposta numero 8, pagina 1
Mi dispiace ma non e come dici tu. L'esercizio e chiaro. La funzione ricorsiva e quella indicata e le funzioni ausiliarie sono
EVENTUALI, quindi possono anche non esserci. Se il tuo professore ti ha detto diversamente, cambia professore.
Gi'a che ci sei, dai al tuo professore la soluzione, che e questa (senza alcuna funzione ausiliare e con la funzione ricorsiva
implementata come richiesto)
Torna su
* il 2014-05-30 enrico.rotundo@studenti.unipd.it ha scritto:
Passa alla risposta numero 9, pagina 1
```

8. Ruoli

Tutti i componenti del gruppo hanno partecipato attivamente alla realizzazione di UniOverflow. Inizialmente si è effettuata in gruppo una progettazione ad alto livello, volta ad individuare tematiche, layout, utenti e componenti a cui si sarebbe rivolto il sito. In seguito si è proceduto con la suddivisione dei ruoli che viene di seguito riportata:

- Giacomo Fornari (1029118): View, Controller, CSS, layout, accessibilità;
- Serena Girardi (1029253): View, Controller, CSS, javascript, validazione;
- Federico Poli (1032418): back-end, classi di Lib, Controller, Middleware;
- Enrico Rotundo (1008052): back-end, classi Model, Lib::XMLCRUD, gestione dati XML, schema XSD, logo.