

Friday

l'innovativo gestore di liste della spesa

Membri del gruppo

- Bosetti Tommaso 185286
- Chiari Sebastiano 185858
- Remondini Leonardo 184563
- Toniolli Marta 187839

Scelte implementative

Durante la realizzazione del progetto Friday abbiamo dovuto fare molte scelte implementative per realizzare le varie funzionalità del sito. In alcune casi queste scelte si sono rivelate efficaci, in altri meno.

In particolare nelle fasi iniziali del progetto abbiamo scelto di non usare i JavaBean perché se è vero che garantiscono un maggiore livello di formalità nella scrittura del codice e in generale nella gestione degli oggetti, nel nostro caso questo livello di formalità ci avrebbe reso più difficoltoso gestire i vari oggetti che gestiamo nel codice.

Allo stesso modo, per quanto riguarda filtri e listeners abbiamo scelto di non usarli. Abbiamo gestito l'aggiornamento "in live" delle informazioni all'interno della stessa pagina tramite web socket e servlet e parametri di sessione. Tuttavia ora ci stiamo rendendo conto che per effettuare controlli sugli indirizzi navigabili all'interno del sito sono necessari i filtri, quindi stiamo cercando una soluzione alternativa al loro utilizzo che sicuramente risulta meno elegante ma che almeno ci permette di non stravolgere la struttura del progetto.

Implementazione

Per quanto riguarda l'implementazione effettiva, ad oggi manca ancora la geolocalizzazione funzionante, perché ci siamo resi conto che la sua implementazione creava molti problemi al progetto e quindi abbiamo deciso di pubblicare il progetto senza questa parte e di implementarla in seguito entro la presentazione. Abbiamo inoltre deciso di creare un branch apposito per questa funzionalità, visto che lavorando in locale la geolocalizzazione impediva l'esecuzione del progetto e quindi la possibilità di implementare altre funzionalità importanti.

Le altre funzionalità richieste dalla descrizione del progetto invece sono state tutte implementate. È da segnalare il fatto che alcune ancora non sono perfette o per via di bug che ancora non riusciamo a gestire o per via di soluzioni programmatiche non propriamente eleganti.

Per esempio, il sistema per diventare Admin e quindi poter creare nuovi prodotti per tutti gli utenti e nuove categorie di prodotti e nuove categorie di lista non è un sistema adatto ad un progetto del genere, però abbiamo ritenuto importante inserire questa funzionalità anche se attualmente il sistema non è sicuro come invece dovrebbe essere. Ovviamente, se questo fosse un progetto reale e noi fossimo un gruppo di sviluppo, questa sarebbe solo una prima release e nelle successive implementeremmo sicuramente un sistema più sicuro e controllato per diventare amministratori di Friday.

Nella struttura del codice abbiamo cercato di rispettare il pattern MVC utilizzando sempre i DAO per connetterci al database e per rispettare la separazione tra business logic e la parte di presentazione. Siamo abbastanza convinti di esserci riusciti, ma siamo anche consapevoli che forse certe soluzioni adottate per rispettare questo pattern non sono esattamente quelle che ci si aspetterebbe in un progetto del genere.

Forse da questo punto di vista la scelta di non usare i JavaBean ci ha un po' messo in difficoltà in queste fasi avanzate del progetto.

Una cosa infine che ancora non è supportata è l'effettivo caricamento delle immagini. Infatti nel nostro programma è inserita la possibilità di inserire un avatar per un utente, o l'immagine di un prodotto o di una lista, ma in realtà nel database viene solo salvato il nome dell'immagine caricata ma non viene effettivamente caricata nelle cartelle del database. Di conseguenza, se l'immagine scelta è una di quelle di default che abbiamo già inserito noi all'interno delle cartelle, allora verrà poi visualizzata correttamente durante la navigazione del sito. Se invece si tratta di un'immagine che non si trova nelle cartelle interne al database, allora non verrà visualizzata.

In generale quindi siamo a conoscenza di diversi bug o funzionalità mancanti che andrebbero implementate e come già detto sopra, se si trattasse della realizzazione di un progetto reale e non fosse solo un progetto universitario, allora questa versione sarebbe solo una prima release e nelle successive implementeremmo tutte le funzionalità mancanti e perfezioneremmo quelle già esistenti cercando di non rendere più complessa un programma che lo è già molto.

Come database abbiamo utilizzato MySQL Workbench perché che lo avevamo già utilizzato nel corso di Basi di Dati e quindi siamo stati in grado di usarlo abbastanza bene.

Utilizzo

Per prima cosa è bene inizializzare il database, creando quindi un server locale all'interno del programma e specificando come porta la 3306, come nome utente root e come password root81097. Dopodiché, una volta creato il server, bisogna importare il file Database.sql che contiene anche il *create schema*. Una volta fatte queste operazioni preliminari, il database dovrebbe essere accessibile.

Una cosa importante da segnalare è che attualmente nel database esiste un utente Admin proprietario di tutte i prodotti e di tutte le categorie di lista che però non è accessibile nel nostro sito web, in quanto la sua password non è criptata e quindi il relativo login fallirà ogni volta. Abbiamo tentato di risolvere questo dettaglio prima della consegna ma per qualche motivo questo fatto creava grossi problemi al funzionamento del sito. Pertanto abbiamo deciso intanto di consegnare una prima versione con ancora questo utente nel database, ma contiamo per la presentazione di eliminare questo problema.

Comunque, come richiesto nel database sono inseriti 10 utenti tutti correttamente registrati. Le loro credenziali di accesso si trovano nel file "credenziali utenti registrati.txt" e sono un insieme abbastanza eterogeneo di amministratori, utenti senza liste, utenti con liste di loro proprietà e/o liste condivise con altri utenti e utenti con solo liste condivise.

Da segnalare inoltre che nel database esiste un utente aaa@cercasi.cercasi che non è inserito nel file .txt di cui sopra perché in realtà è utente che abbiamo usato per fare vari test e che abbiamo lasciato nel database per sicurezza per continuare a fare questi test senza "rovinare" gli altri account.

IMPORTANTE

È importante fare riferimento al branch master per avere delle versioni funzionanti del programma, in quanto negli altri branch non garantiamo che tutte le funzionalità sia implementate e/o funzionanti, perché avendo la possibilità di lavorare sul progetto fino alla sua presentazione, siamo in continua fare di programmazione.