

MovieRatings: Utilizabilitate Web 2.0

Adrian Buzgar

Universitatea "Alexandru Ioan Cuza"

Facultatea de Informatică
Str. General Berthelot, nr. 16, 700483,
Iași
adrian.buzgar@infoiasi.ro

Simona Lazarovici

Universitatea "Alexandru Ioan Cuza"

Facultatea de Informatică
Str. General Berthelot, nr. 16, 700483,
Iași
simona.lazarovici@infoiasi.ro

REZUMAT

Această lucrare prezintă un studiu de caz asupra punerii în practică a unor principii și noțiuni de utilizabilitate într-o aplicație Web, în contextul tendințelor Web 2.0 și a tehnologiilor de ultimă generație folosite pentru dezvoltarea aplicațiilor de tip RIA. Vom evidenția aceste aspecte pornind de la o prezentare a proiectului MovieRatings (MOV R).

Cuvinte cheie

RIA, Web 2.0, SOA, servicii Web, details-on-demand, interfață utilizator *zoomable*.

Clasificare ACM

H5. Information Interfaces and Presentation.

INTRODUCERE

Conturarea principiilor Web 2.0, și mai ales evoluția rapidă a tehnologiilor RIA (Rich Internet Application) [1], permit realizarea unor aplicații Web cu o mai mare libertate creativă în construirea interfețelor, ducând la o interactivitate sporită și la adoptarea facilă a unor principii noi de utilizabilitate.

MovR este o aplicație Web care oferă cineaștilor posibilitatea de a vizualiza topurile celor mai bine cotate filme dintr-o bază de date multimedia, în același timp permițându-le modificarea ierarhiei prin propriile evaluări. În plus, utilizatorii pot adăuga conținut și pot adnota semantic sursele de date pentru a eficientiza căutările în funcție de diferite criterii.

Am realizat acest proiect pentru a ne familiariza cu o serie de tehnologii, prezentate în următoarele secțiuni, și pentru a verifica măsura în care se poate realiza facil și rapid o interfață utilizator superioară celor obținute folosind abordarea clasică XHTML & CSS.

PRINCIPIILE WEB 2.0

În pașii incipienți din dezvoltarea unei aplicații de tip RIA precum MovR, este utilă formarea acelui subset de principii Web 2.0 [2] care facilitează atingerea scopurilor finale. Am considerat că proiectul nostru se identifică cel mai bine cu următoarele principii:

- *Utilizatorii adaugă conținut (Users add value)*: proiectul MOV R își dezvoltă mereu baza de date folosind contribuțiile utilizatorilor (evaluări ale filmelor existente, noi filme, adnotări).
- *Prelungirea stadiului beta al unei aplicații Web (Perpetual Beta)*: în cazul aplicațiilor Web de ultimă generație se evită intenționat noțiunea de produs final. Adaptarea dinamică la reacțiile și noile cerințe ale utilizatorilor oferă flexibilitate crescută.
- *SOA (Service Oriented Architecture)* [3]: Aplicațiile Web beneficiază de modularizarea arhitecturii în mod distribuit, componentele comunicând în special prin

servicii Web. Diferitele subsisteme ale MOV R sunt conectate slab (*loosely coupled*) prin intermediul serviciilor expuse și permit extinderea și reutilizarea.

- *Cooperarea în locul controlului*: MOV R folosește pentru întreg conținutul multimedia surse externe de date: servicii Web ale Flickr și YouTube, resurse ale Internet Movie DataBase (IMDB). Toate informațiile interne sunt stocate în format XML și sunt accesibile pentru alte aplicații prin intermediul unui API (*Application Programming Interface*) ușor de folosit.

ARHITECTURA MOV R

Pornind de la principiile enumerate mai sus, MovR are o arhitectură multistratificată distribuită, ale cărei module sunt reprezentate în figura 1.

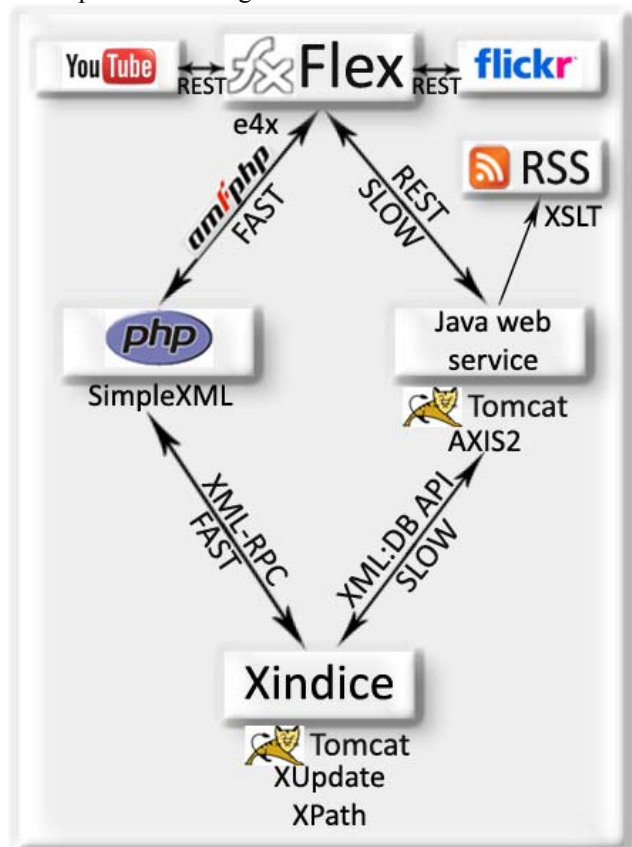


Figura 1. MovR – arhitectura distribuită.

Spre deosebire de aplicațiile Web tradiționale (thin-client), care pun la dispoziția mașinii client o interfață prefabricată și lasă logica aplicației în seama serverului, MovR folosește Adobe Flex 3.0 pentru a concentra majoritatea proceselor într-un thick-client, minimizând sarcina de lucru a serverelor, indiferent de complexitatea (în acest caz vizuală) a aplicației. Astfel, cea mai mare parte a codului a fost scrisă în ActionScript 3.0, acesta oferind flexibilitatea RIA într-un limbaj orientat-obiect [4].

Trebuie menționat totuși faptul că acest avantaj are un cost, acela ca un browser nu poate rula un thick-client RIA în lipsa unui plugin specific fiecărei tehnologii de acest tip (Flash Player pentru Adobe Flex, Java Runtime Environment în cazul JavaFX sau Silverlight Player în cazul Microsoft Silverlight).

Partea de prezentare a aplicației noastre recurge la serviciile oferite de Flickr și YouTube (făcând apel la API-uri publice REST [5]), pentru a atașa automat conținut multimedia datelor, și interoperează cu modulele middleware ale MovR pentru a prelua și actualiza informațiile din sistem.

Există două modalități de acces la date. Prima, mai rapidă, este implementată în PHP, pune datele la dispoziție prin intermediul AMFPHP (format binar) și le obține prin protocolul XML-RPC (precursorul mai rapid al SOAP). A doua modalitate, mai lentă, folosește servicii Web (pe baza paradigmei REST) dezvoltate în Java și încărcate pe un server Apache Axis 2, care pot oferi, la cerere, ultimele modificări ale bazei de date în format RSS.

Baza de date este de tip XML nativ (Apache Xindice [6]), lucrând cu colecții de documente XML care conțin elemente DCMI (Dublin Core Metadata Initiative [7]). Accesarea și modificarea se face folosind vocabularele XPath, respectiv XUpdate.

Una dintre consecințele alegerilor din faza de modelare a arhitecturii a fost reducerea traficului dintre client și server. Aplicația trebuie descărcată o singură dată, după care tot conținutul dinamic este prelucrat local. Imaginile sunt descărcate din surse externe, singura comunicare cu serverele ținând de transportul informației XML în format binar AMF (Action Message Format), eventual de apelul la serviciile Web.

INTERACȚIUNE SI UTILIZABILITATE

O noțiune importantă în cadrul interacțiunii om-calculator este utilizabilitatea (usability), definită prin măsura în care un produs poate fi folosit de către utilizatori pentru a îndeplini anumite cerințe cu eficiență, eficacitate și satisfacție [8], într-un anumit caz de utilizare.

Eficacitatea (effectiveness) presupune acuratețea cu care un sistem reușește să efectueze operațiile cerute, precum și acoperirea tuturor aspectelor vizate. Eficiența (efficiency) este un indicator al efortului pe care un utilizator trebuie să-l depună pentru a-și îndeplini scopurile. Satisfacția utilizatorilor poate fi cuantificată prin aprecierile acestora în urma interacțiunii cu un anumit sistem.

În practică, datorită naturii subiective a aprecierii utilizabilității unei aplicații, există mult mai multe principii (și clasificări) după care se ghidează realizarea interfeței utilizator. Printre acestea o influență importantă asupra interfeței MovR a avut-o noțiunea de navigare continuă: perspectiva utilizatorului se modifică fluent, evitând tranzițiile bruște, adică schimbarea întregului conținut vizibil (așa cum se întâmplă în cazul aplicațiilor Web cu o interfață utilizator modelată în XHTML).

Două tehnici legate de interacțiune pe care am încercat să le integrăm în proiect sunt details-on-demand („detalii la cerere”) și zoom [9]. Prima presupune afișarea detaliilor numai atunci când utilizatorul cere explicit acest lucru, reducând încărcarea suprafeței vizibile cu informații. A doua este în strânsă legătură cu noțiunea de interfață zoomable, un mediu în care accesul la amănunte se face

prin scalarea anumitor obiecte grafice, în care vor apărea noile date. Combinarea celor două tehnici are un impact pozitiv asupra experienței utilizatorului și are ca urmare reducerea consumului de resurse și a traficului nesolicitat, mai ales la pornirea aplicației. Pe de altă parte, această abordare poate ridica probleme în implementare, atunci când tehnologiile alese nu se pretează la efecte grafice ca scalarea.

În cele ce urmează, ne vom folosi de unele scenarii de utilizare ale aplicației MovR pentru a delimita contextul în care sunt folosite principiile și tehnicile legate de utilizabilitate.



Figura 2. Vedere de ansamblu a interfeței MovR.

Vizualizarea topurilor

Inițial, încărcarea topurilor se realizează printr-o animație care deplasează filmele din centrul suprafeței vizibile spre exterior, rezultatul fiind observabil în figura 2. Topurile conțin numai cele mai bine cotate filme, astfel evitându-se încărcarea întregii baze de date. Pentru fiecare film, informațiile sunt succinte. Se încarcă afișul de pe IMDB și nota corespunzătoare.

Elementele de interfață au libertate de mișcare completă, fiecare putând fi deplasat prin drag and drop, în contrast cu aplicațiile Web uzuale. Pentru a înlesni adaptarea rapidă a utilizatorilor, majoritatea posibilităților de interacțiune sunt prezentate prin intermediul unor mesaje de ajutor (tooltip-uri).

Accesul la detalii

Pentru a obține informații detaliate legate de un film din topuri, utilizatorul efectuează operația de zoom, mărind afișul filmului respectiv cu roțița mouse-ului până la un anumit prag, după care suprafața vizibilă a filmului se transformă, făcând loc detaliilor adiționale: anul producției, regizorul, genurile la care aparține filmul, intriga, actorii etc.

De asemenea, prin intermediul legăturii cu Flickr, sunt încărcate patru imagini relevante, în format mic (thumbnails), care pot fi mărite prin aceeași modalitate ca și afișul filmului.

Tot în această perspectivă apar butoanele pentru redirecționarea către trailer-ul filmului de pe YouTube și către pagina filmului de pe IMDB (singurele excepții de la regula navigării continue), cât și pentru funcționalitatea de adnotare.

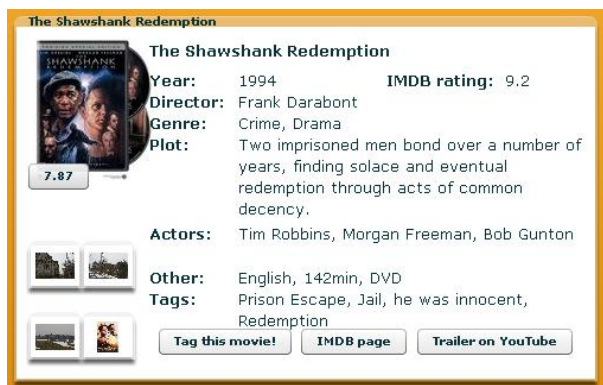


Figura 3. Vedere detaliată.

Accesul la unelte

Un criteriu central în implementarea interfeței utilizator a fost minimalismul, acesta presupunând afișarea în orice moment a strictului necesar de elemente. Există o singură zonă de instrumente, așezată inițial în centrul aplicației, care poate fi deplasată conform preferințelor utilizatorului. Aceasta zonă conține un câmp pentru căutare, precum și butoane pentru adăugare de filme, vizualizarea întregii baze de date și resetarea perspectivei.

În situațiile în care anumite funcționalități necesită introducerea informațiilor în sistem, zona de unelte se redimensionează printr-o animație, butoanele și câmpurile noi necesare fiind adăugate dinamic, ca în figura 4. Odată cu acestea, apare și un buton de anulare a operației începute, acționarea acestuia având efectul de revenire la forma inițială.

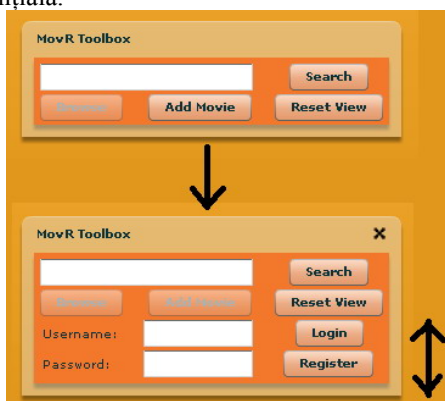


Figura 4. Redimensionarea zonei de unelte pentru autentificare.

Datele introduse în MovR sunt mereu validate. Fiecare tip de eroare găsită determină apariția unui mesaj personalizat, modal, ca în figura 5. Grijă pentru aceste detalii aduce beneficii importante: duce la evitarea situațiilor în care aplicația poate eșua (eventual în urma unui atac de tip cross-site scripting [10]), la eficientizarea interacțiunii cu sistemul prin eliminarea confuziilor, și este în același timp o formă perceptibilă de respect față de utilizatori.

Am căutat să integrăm cât mai multă funcționalitate fără să cerem autentificarea utilizatorului, aceasta fiind necesară doar la modificarea informațiilor din sistem, adică adăugarea, evaluarea sau adnotarea prin tagging a filmelor.

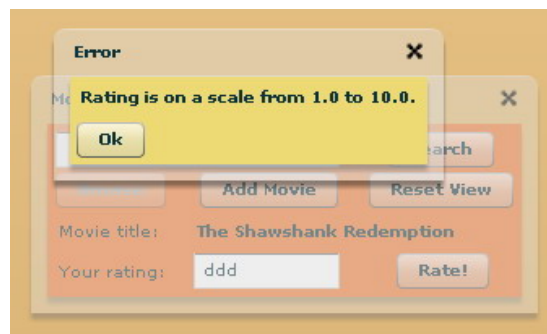


Figura 5. Mesaj de eroare.

Autentificarea

Nu există un buton separat pentru autentificare în zona de unelte, această operație având loc automat numai atunci când este necesară. Dacă s-a încercat pornirea unei operații care necesită autentificarea, cum ar fi evaluarea unui film, un mesaj de eroare indică acest fapt, moment în care utilizatorul poate alege să accepte procesul (Ok), caz în care zona de unelte se redimensionează consecutiv pentru login și imediat după aceea pentru operația cerută. Odată intrat în sistem, numele utilizatorului apare în bara de titlu a zonei de unelte.

Un avantaj al aplicației MovR este acela al simplității cu care utilizatorii noi se pot înregistra în baza de date. Folosirea în mod obișnuit a formularelor XHTML pentru interfața utilizator creează vulnerabilitate față de atacuri care vizează înregistrarea automată a unui număr foarte mare de utilizatori. Soluțiile actuale de prevenire a acestor situații (imagini captcha, email-uri de confirmare) pot deveni agasante pentru utilizatori. În cazul MovR, întreaga aplicație fiind distribuită în format binar (.swf), acest tip de vulnerabilitate este mult mai greu de exploatat. În consecință, tot procesul de înregistrare sau autentificare se rezumă la introducerea numelui și a parolei.

Căutarea

În zona de unelte este integrată facilitatea de căutare. Se poate folosi o gamă variată de criterii de căutare: adnotări, titlu, actori, regizor, gen sau descriere.

În MovR, rezultatele căutării sunt evidențiate direct în topuri printr-o animație care are ca efect micșorarea prin scalare a elementelor. Factorul de scalare pentru filmele găsite este 0.8, în timp ce pentru celelalte este 0.3, diferențierea fiind vizibilă în figura 6. Această abordare contribuie la dinamismul vizual al aplicației.



Figura 6. Rezultatele căutării.

Evaluarea filmelor

Fiecare utilizator poate evalua filmele din baza de date, modificând în timp real topul preferințelor. Operația necesită autentificarea, pentru că odată introdusă nota pentru un film, aceasta poate fi modificată ulterior de către utilizatorul respectiv.

Orice modificare a notelor duce la actualizarea automată a topurilor, aceasta însemnând o animație prin care filmele vizibile în aplicație își schimbă coordonatele în funcție de poziția recalculată în top. Este posibil ca unele filme să dispară din suprafața vizibilă, moment în care locul lor este luat de noile intrări în top.

Situația topurilor este actualizată dinamic, adică influența unui utilizator produce feedback rapid în sesiunea curentă a tuturor celorlalți. În acest fel utilizatorului îi este evidentă nu numai interacțiunea cu aplicația, ci și cu întreaga comunitate de cinefili care o folosesc.

Resetarea perspectivei

Faptul că toate elementele vizuale prezente în aplicație pot fi deplasate și redimensionate oferă avantaje din punct de vedere al interactivității, dar în același timp poate crea senzația de aglomerare. Singurul indicator al poziției unui film într-un top fiind tocmai amplasarea pe ecran, această informație se pierde când utilizatorul îi schimbă filmului coordonatele.



Figura 7. Aglomerare vizuală.

Pentru a restabili perspectiva inițială se poate folosi butonul de resetare din cadrul zonei de unelte. Această funcție pornește o animație care scalează și re poziționează elementele grafice, schimbând unde este necesar starea vizuală a unui film din modul detaliat în cel succint, efectuând practic tranziția fluidă de la figura 7 la figura 2.

CONCLUZII

În cadrul acestui proiect am încercat să înglobăm o serie de principii ale filosofiei Web 2.0 și de noțiuni, sperăm

inovatoare, care țin de utilizabilitate. Rezultatul actual dovedește faptul că, în cazul MovR libertatea de exprimare vizuală și interacțiunea nu mai sunt limitate tehnologic, ci doar de experiența și imaginația implementatorilor. Acest fapt se datorează exclusiv noilor tehnologii Web folosite în dezvoltarea aplicațiilor de tip RIA, printre care se numără și Adobe Flex (integrată în suita Adobe AIR).

Deși intuitiv costurile în materie de performanță și trafic de rețea ale unei aplicații RIA sunt mari, depășirea micilor dificultăți legate de integrarea modulelor proiectului MovR (în special în cazul interoperabilității Flex – PHP și PHP – Xindice) a dovedit contrariul.

Nu am dorit să fim partizanii unei singure inițiative din domeniu, alegerea noastră fiind motivată doar de succesul unui proiect anterior. Considerăm că orice altă tehnologie similară, cum ar fi Microsoft Silverlight sau JavaFX, poate da rezultate la fel de bune odată ce se trece de bariera deprinderii facilităților oferite.

În principal din perspectiva utilizabilității aplicațiilor Web 2.0, considerăm că tehnologiile RIA aduc suficiente beneficii încât să-și motiveze pe deplin integrarea în proiecte ca MovieRatings, mai ales pentru dezvoltarea interfețelor utilizator.

REFERINȚE

1. Jeremy Allaire, *A Next Generation Rich Client*, Macromedia Whitepaper, 2002.
2. Tim O'Reilly, *Design Patterns and Business Models for the Next Generation of Software*, 2005, <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
3. The Open Group, *Service Oriented Architecture*, <http://www.opengroup.org/projects/soa/>
4. Adobe Flex, <http://labs.adobe.com/technologies/flex/>
5. Roy Fielding, *Architectural Styles and the Design of Network-based Software Architectures*, www.ics.uci.edu/~fielding/pubs/dissertation/top.htm
6. Apache Xindice, <http://xml.apache.org/xindice/>
7. Dublin Core Metadata Initiative, <http://dublincore.org/>
8. ISO 9241-11, *Guidance on Usability*, 1998, ISO.
9. Shneiderman, Plaisant, *Designing the User Interface*, 4th Edition, Addison-Wesley, 2004.
10. WhiteHat Security, *Cross-site Scripting*, 2006. www.whitehatsec.com/downloads/WHXSSThreats.pdf