Sistem de regăsire a informaţiilor

Paul-Daniel Iurea

Rezumat

Sistemul de regăsire a informațiilor este un proiect dezvoltat sub forma unei aplicații web. Aplicația își propune strângerea în același loc a mai multor surse de informație (știri), oferind posibilitatea utilizatorului de a sorta informația după criterii alfabetice, după tipul informației (știri din sport, știri din domeniul tehnic, știri din domeniul financiar etc) sau de a afișa doar de la o sursă de știri specifică existentă în sistem. Oferă de asemenea facilitatea de a salva un istoric al știrilor și a-l vizualiza mai târziu și de a face căutare în întregul sistem după cuvinte cheie pentru regăsirea mai ușoară a unor informații specifice. Sursele de informație sunt adăugate în sistem după dorința utilizatorului, prin copierea link-ului RSS al website-ului de știri dorit în formularul de adăugare din cadrul aplicației.

Decizia de a dezvolta un asemenea sistem a apărut din dorința de a concentra sursele de informare frecventate, eliminând în acest mod orice altă distragere a rețelei Internet, atunci când se realizează o căutare pe Google sau pe orice alt site web care folosește un motor de inferențe. În felul acesta se eficientizează timpul alocat informării, oferind utilizatorului șansa de a afla cât mai multe într-un timp fără distorsiuni. Au fost alese tehnologiile prezentate în continuare pentru a se pune în aplicare cunoștințele acumulate în cadrul acestora în decursul anilor de facultate și pentru șansa de a le îmbogăți.

Tehnologiile folosite pentru implementarea aplicației sunt:

- Symfony2 (framework OOP PHP);
- Doctrine2 (framework ce conține două biblioteci PHP - ORM <Object Relational Mapper>, respectiv DBAL <Database Abstraction Layer>, framework realizat după conceptele implementate în Hibernate și adaptat pentru limbajul PHP);
- Baze de date MySql;
- Jquery 1.11.1;
- Bootstrap 3.3.4;
- JAVA;

Aplicația este capabilă să adauge în sistem surse de informare (link-uri RSS), să afișeze descrierea articolelor, titlul, sursa, data publicării acestora și poza de prezentare a articolelor unde există. Oferă facilități de paginare, odată depășită capacitatea de 7 articole pe pagină, adăugând posibilitatea de a naviga la pagina dorită. Numărul maxim de pagini din care utilizatorul poate alege fără a selecta opțiunea de a vedea mai multe pagini este 25. Totodată aplicația oferă posibilitatea de a naviga pe pagina imediat următoare sau pe cea precedentă sau poate naviga direct peste cele 25 de pagini curente pentru a avea posibilitatea de a vedea în continuare și alte pagini.

Sunt implementate funcționalități de sortare alfabetică, afișare după tipul informației și afișarea doar dintr-o sursă specifică, aleasă din cele existente în sistem.

Toate cele menționate sunt implementate în Symfony2.

Modulul de căutare este implementat în Java. Algoritmul de căutare se bazează pe un crawler web. Modulul programat în Java primește ca date de intrare un vector de link-uri RSS, și se ocupă de parsarea fișierelor XML aflate în spatele acestor link-uri.

Acest modul își propune se implementeze ideea algoritmului MapReduce.

Într-o primă etapă, URL-urile articolelor găsite în RSS-uri sunt indexate direct (mapate), la fel și cuvintele din articole. Titlurile acestora, respectiv cuvintele sunt transformate prin funcția hash MurmurHash în varianta numerică pentru o extragere mai rapidă la căutare. Cuvintele sunt tranformate numai după ce sunt eliminate acele cuvinte de legătură (prepoziții, conjuncții), după ce sunt verificate drept aparținând sau nu unei liste de excepții (nume proprii) și după ce sunt trecute

printr-un dicționar de cuvinte care le întoarce forma originală, nearticulată.

Indecșii direcți sunt salvați în fișiere text.

În a doua etapă are loc procedura de reducere, fiind creat și un al treilea fișier în care se regăsesc codurile hash ale titlurilor articolelor, alături de codurile hash ale cuvintelor găsite în aceste articole și numărul de apariții al acestora în cadrul articolului. După crearea acestui fișier, reprezentând index-ul direct, are loc crearea index-ului invers, acest lucru însemnând crearea unui fișier text care să conțină pe prima poziție codul hash al cuvântului, la care se adaugă toate codurile hash ale link-urilor tuturor articolelor în care acesta se regăsește alături de numărul de apariții al cuvântului din cadrul articolului.

Pentru operația de căutare are loc prima etapă asupra cuvintelor din interogare, respectiv codificarea acestora cu algoritmul hash MurumurHash, după care se extrag titlurile articolelor ce conțin cuvintele căutate, în ordinea descrescătoare a numărului de apariții pentru fiecare cuvânt și sunt transmise ca date de ieșire pentru modulul dezvoltat în PHP, Symfony2.

Aplicația dispune de un modul de salvare a istoricului și de a afișare a tuturor știrilor salvate pe server. Salvarea se realizează printr-un proces automatizat, link-urile din baza de date fiind verificate o dată la o perioadă de timp cu ajutorul unui cronjob înscris într-un crontab al sistemului de operare Linux. Cronjob-ul apelează o comandă care are rolul de a salva fișierele RSS ale surselor înregistrate în sistemul aplicației, doar dacă se găsesc diferențe ale datei și orei apariției între fișierul RSS aflat pe website-ul sursei directe și fișierul RSS stocat local. Pentru afișarea arhivei, aplicația consultă directorul „history” aflat în directorul public. Utilizatorul are posibilitatea atât de a opri salvarea istoricului și de a reporni această facilitate ori de câte ori își dorește, cât și de a șterge arhivele salvate.

Primul capitol al lucrării prezintă o introducere în sisteme de regăsire a informației, motivând construirea unui asemenea sistem.

Capitolul al 2-lea prezintă o serie de fundamente teoretice și face referire la documentarea bibliografică privind tehnologiile utilizate pentru dezvoltarea sistemului, funcționarea cronjob-urilor în sistemul de operare Linux, algoritmii implementați, funcționarea roboților Web și transmiterea datelor prin socket-uri.

Capitolul al 3-lea reprezintă modul în care a fost proiectat sistemul, prezentând modulele componente și legăturile realizate între acestea și ilustrând prin diagrame UML clasele din care sunt formate modulele.

În capitolul 4 se prezintă modul de implementare a componentelor ilustrate în capitolul al 3-lea, alături de mediile de dezvoltare utilizate și dificultățile întâmpinate pe parcursul dezvoltării și se explică succint funcționarea interfeței cu utilizatorul.

Atât în capitolul 3, cât și în capitolul 4 se prezintă modul de proiectare și de implementare pentru o serie de comenzi utilizabile atât în terminalul sistemului de operare Linux, sistem sub care a fost dezvoltată aplicația, cât și în liniile de comandă ale altor sisteme de operare cunoscute ce ar putea găzdui aplicația.

În capitolul 5 sunt prezentate rezultate experimentale privind încărcarea sistemului în diferite navigatoare (eng. *browsere*) și teste privind anumite ramuri ale funcționării aplicației.

Utlimul capitol prezintă concluziile trase după implementarea și testarea sistemului și posibile direcții pentru viitoare dezvoltări.

La finalul lucrării pot fi găsite anexele ce conțin codul claselor create, structurat pe module și modul de utilizare a comenzilor Linux implementate.