

Aplicatie web pentru realizarea unei biblioteci on-line

Prep. Dragos CRISTEA

Catedra de Contabilitate si Informatica Economica, Universitatea "Dunarea de Jos" Galati

This paper presents the main techniques and methods involved in the implementation of an Web application by presenting a suggestive example: the construction of an on-line library. We can find many such libraries on the Internet but all share the same problem: the users can only browse the information but they cannot upload any paper. So we have extended the classic model by given the permission to upload files into the file server in a controlled manner by providing the adequate security credentials. For this we present the software that it's involved in the construction of the example and why we choose some products when there are many similar existent on the market. The application shows the basic steps involved in its construction and how it's possible to work with great amounts of data without affecting its functionality.

Keywords: web application, database systems, vbscript, client-server, forms, asp.

Aplicatia dezvoltata este structurata pe doua componente: o biblioteca online si o extensie a celei dintâi, fiind reprezentata de un sistem de comunicare între membrii institutiei (în cazul de fata profesori si studenti). Ambele componente au ca suport de stocare a informatiei o baza de date, fiecare dintre ele este alcatuita dintr-o colectie de pagini Web, *html* respectiv *asp*.

Baza de date a fost realizata în MSSql, structura ei tabelara este relativ simpla, respectând regulile normalizarii. Ea contine cinci tabele:

- Tabela **AddU** foloseste la memorarea datelor despre utilizatorii obisnuiti ai aplicatiei, adica a acelor persoane care nu au decât drepturi de cautare si nu de adaugare sau de administratori. Atributele sunt: Name, Parola, Nume, Prenume, Telefon, Mail, Adresa www, Nivel. Name este cheia primara în aceasta tabela. Restul atributelor, mai putin "**nivel**", formeaza "cartea de vizita" a utilizatorului. Site-ul "Biblioteca" ofera posibilitatea cautarii utilizatorilor simpli si a vizualizarii datelor introduse de acestia. Atributul "**nivel**" este mai special, în sensul ca el asigura protectia tuturor paginilor din site. Orice utilizator din tabela AddU are implicit nivelul 0. Aceasta înseamna ca o parte a paginilor site-ului pot fi de asa natura protejate încât daca o persoana stie adresa unei pagini ea sa nu o poata accesa decât daca s-a autentificat în prealabil si a fost creata pe server o variabila aferenta în care nivelul utilizatorului corespunde cu nivelul de la care se poate avea acces în pagina respectiva.

punde cu nivelul de la care se poate avea acces în pagina respectiva.

- Tabela **AddUser** este o "sora" mai mare a tabelii precedente în sensul ca ea cuprinde în mare parte aceleasi atribute, plus câteva specifice. Aceasta este tabela acelor utilizatori care au dreptul de a scrie în baza de date în tabela Lucrare, cu alte cuvinte au dreptul de a adauga informatii prin operatia de upload pe server, informatii care ulterior pot fi consultate prin download. Nivelul implicit al utilizatorilor din aceasta tabela este 1. Nivelul 1 nu permite accesarea paginilor cu accesul 0, pentru aceasta toti utilizatorii trebuie sa-si faca un cont primar în tabela AddU. Înregistrările în aceasta tabela se fac de catre administratorul bazei de date, pe baza de buletin de identitate. El va da celui care solicita un astfel de cont un cod special, complet confidential, care reprezinta în baza de date si cheia primara, acest cod fiind necesar pentru introducerea de înregistrari în tabela "Lucrare". Principiul care sta la baza acestei strategii este faptul ca atributul Cod_Ut, devenit cheia secundara în tabela "Lucrare", nu va permite în aceasta tabela decât generarea de înregistrari care au un **cod_ut** valid. Codul utilizatorului va fi ca o semnatura a acestuia pentru ceea ce trimite în baza de date. Tocmai din acest motiv se recomanda pastrarea secreta a acestui cont. Câmpul **Numeut** defineste numele utilizatorului, la fel ca si în tabela precedenta, scriptul

VB care se va ocupa de transmiterea informatiei catre baza de date va verifica daca acesta este unic, în caz contrar va fi necesara alegerea altui utilizator. Restul atributelor sunt aceleasi ca si în tabela "AddU".

- Tabela **Lucrare** se foloseste pentru retinerea înregistrarilor care reprezinta informatia trimisa pe server. Atributele acestei tabele sunt: **L_Id, Cod_Ut, Titlu, Subiect, Nr_Pagini, Nume fisier, CodLucrare, Nume**. În primul rând se precizeaza cheia primara: **CodLucrare**. La fel ca si **Cod_Ut**, valoarea acestui atribut este strict secreta deoarece cu ajutorul lui se pot realiza o serie de operatii asupra unei înregistrari: modificare si stergere. Fiecare utilizator primeste la înscriere cinci astfel de coduri pe care le va folosi când va introduce o înregistrare în baza de date. Prezenta cheii secundare "**Cod-Ut**" asigura faptul ca nici un articol nu va putea fi introdus în aceasta tabela fara codul personal care a fost definit în tabela "AddUser". "**Nume fisier**" vizeaza fisierul care va fi trimis pe serverul de fisiere. Restul atributelor, ("**Titlu, Subiect, Nr-pagini**") sunt caracteristici ale lucrarii, necesare cautarii cât mai precise a acesteia.

- Tabelele **Profesor** si **Informatii** sunt legate prin intermediul cheii primare NumeUser. Rolul acestor doua tabele este în acest context unul strict educational: fiecare profesor dintr-o facultate își poate crea un cont cu anumite caracteristici asemanatoare unui utilizator si ulterior pe baza acestui cont el poate lasa diferite informatii pentru studenti, acestia pot la randul lor, cunoscând numele profesorului, sa cunoasca informatia pe care acesta a dorit sa le -o comunice.

✚ *Structura bibliotecii on-line*, prima componenta a aplicatiei contine urmatoarele pagini.

Pagina **index.html** contine o scurta prezentare a ceea ce face aplicatia, ea fiind realizata numai în cod HTML.

Pagina **intro.html** contine doua trimiteri: una la o pagina de creare a unui utilizator nou, în cazul în care persoana care acceseaza nu are definit unul, sau catre o pagina de autentificare **logincauta.asp** care va cere utilizatoru-

lui un nume de user si o parola pentru a putea continua.

Paginile de adaugare utilizator respectiv autentificare sunt indispensabile unei astfel de aplicatii. Ele înglobeaza scripturi scrise în limbajul VBscript care, în primul caz, colecteaza datele introduse în formularul HTML si le transmite catre baza de date, iar în cel de-al doilea verifica daca în baza de date exista un utilizator cu numele si parola identice cu cele introduse de utilizatorul care doreste sa foloseasca aplicatia.

Rolul paginii **mainmenu** este de cuprins pentru întreg site-ul. De aici un utilizator poate avea acces la toate functiile aplicatiei, mai puțin la cele de administrative. Trebuie precizat ca exista mai multe tipuri de utilizatori: care au numai dreptul de a cauta în articolele stocate în baza de date si de a face download la documentatie, care au dreptul de adauga articole si administratori cu drepturi totale.

Pagina de cautare (cauta.html) are ca principala functie ajutarea utilizatorului de a cauta o anumita lucrare sau numele unui utilizator care a depus o lucrare. Pentru a cauta un autor este suficient a furniza una sau mai multe din urmatoarele informatii: *numele userului, numele de familie, prenume, telefon, mail*. În cazul cautarii unei lucrari sunt necesare urmatoarele date: *identificatorul lucrarii, titlu, subiect, utilizator, numele fisierului*

Paginile care permit preluarea datelor introduse de utilizator se numesc "cautaautor.asp", respectiv "cautalucrare.asp". Scopul acestor doua pagini este de a furniza scripturilor din paginile "rezultateautor.asp" si "rezultatelucrare.asp" parametrii necesari cautarii. Functionarea acestor pagini se face pe baza principiilor interogarii cu parametri. Fie urmatoarea declaratie SELECT:

```
SELECT Numeut, Nume, Prenume, Telefon, Mail, Adresawww
FROM dbo.AddUser
WHERE Numeut LIKE 'varnumeut' AND
Nume Like 'varnume' AND Prenume
Like 'varprenume' AND Telefon Like
'vartelefon' AND Mail Like
'varmail'
```

Varnumeut, varnume, varprenume, vartelefon si varmail reprezinta parametri care sunt introdusi în pagina "cautaautor.asp" de catre

utilizator în câmpurile text aferente. Scriptul din pagina “rezultateautor” preia acești parametri și pe baza lor, a unui recordset și a unei surse de date execută interogarea bazei de date și afișează rezultatele acesteia. Prințo astfel de sursă de date se definește driverul prin care sistemul de operare va comunica cu baza de date. Se pot defini surse de date pentru sisteme Oracle, FoxPro, Acces, Sybase etc.

Pagina de afișare este reprezentată de fișierul **afisare.html**. Odată accesată ea oferă următoarele posibilități de navigare: afișarea tuturor subiectelor distincte din baza de date, dacă utilizatorul va dori să “conducă” o interogare pentru a găsi un articol cu un anumit subiect și afișarea tuturor lucrărilor cu un anumit subiect. Utilizatorului i se cere să introducă subiectul (în pagina lucrari.html) iar pagina **lucrari.asp** va afișa lucrările, afișarea numelui (ales de utilizator) numelui de familie, prenumelui.

În continuare sunt prezentate paginile de administrare, accesibile numai acelor utilizatori care au un anumit nivel în baza de date (respectiv acela de administratori)

Pagina de adaugare. Cu ajutorul acestei pagini se pot introduce în baza de date și în serverul de fișiere diferite lucrări. În momentul în care se dorește adaugarea unei lucrări, utilizatorul este redirectionat către o nouă pagină de autentificare. Diferența față de prima pagină de autentificare este că aici nu vor putea să se autentifice decât acele persoane care au un anumit nivel (1) setat în baza de date. După autentificare este accesată pagina **adaugalucrare.asp**, responsabilă cu introducerea datelor în baza de date. Dacă datele au fost introduse corect, va fi accesată pagina **upload.html** care permite upload fișierelor pe server

Paginile de modificare (modifica.html, modificalucrare.html, modificalucrare.asp) se folosesc pentru schimbarea atributelor unei lucrări.

✚ A două componentă, numită de comunicare, a aplicației are următoarea structură.

Pagina profesor.html conține un meniu cu trimitere la paginile: *catedre.asp*, *vezipro-*

profesor.html, *alegeprofesor.asp*. Pagina *catedre.asp* afișează toate catedrelor din care fac parte profesorii înscrși în baza de date; *veziprofessori.html* permite aflarea tuturor profesorilor care lucrează la o catedră. Se va introduce în prealabil numele catedrei ai căror profesori trebuie identificați (în pagina *professori.asp*), după care rezultatele vor fi furnizate de către pagina *veziprofessori.asp*. Pagina *alegeprofesor.asp* permite alegerea profesorului a cărui informație se dorește a se vizualiza, rezultatele fiind vizibile în pagina *vizualizeaza.asp*.

Introducerea în baza de date a datelor despre profesori precum și informația pe care aceștia vor să o transmită se face prin intermediul a două pagini speciale (*adauga.asp* și *adaugainformatie.asp*), care nu se pot vizualiza accesând meniurile site-ului. Ele sunt vizibile numai utilizatorilor cu nivelul de acces 3 (profesori și administratorii bazei de date).

Datele necesare despre profesori sunt aproximativ aceleași cu cele din cazul unui utilizator obișnuit. Excepție este câmpul “catedră” în care se introduce numele catedrei unde lucrează fiecare profesor. Adaugarea mesajelor în baza de date se face astfel: profesorul introduce în pagina *adaugamesaj.asp* numele informației (de exemplu note, material, bibliografie), numele sau de utilizator, precum și informația propriu-zisă. Apăsarea butonului *Trimite* are ca efect transmiterea informației preluată din formularul html, către baza de date.

Un scenariu de folosire a unui astfel de sistem este prezentat în continuare. Întâi este selectat în pagina *alegeprofesor.asp* numele profesorului sau a informației de vizualizat (în pagina *alegeprofesor.asp*). Pagina rezultată în urma preluării datelor din acest formular, *vizualizeaza.asp* oferă informația găsită în urma interogării bazei de date.

Concluzii

Datorită vitezei exponentiale de dezvoltare a Internetului, abilitățile noastre în a susține această creștere vor depinde în mare măsură de viitoarele aplicații pe care le vom proiecta și implementa. Limitele sistemului prezentat sunt date în primul rând de mediul pentru ca-

re a fost construit, el nefiind conceput pentru gestionarea unui numar mare de utilizatori concurenti în ceea ce priveste trimiterea documentelor catre server. Din acest punct de vedere principalele îmbunatatiri care se pot realiza constau în redefinirea indexurilor din tabelele bazei de date, automatizarea unor operatii prin folosirea cu precadere a procedurilor stocate, implementarea tranzactiilor. Serverul de fisiere este o alta componenta a acestei aplicatii care poate, în cazul unui flux mare de informatii, sa genereze o serie de restrictii privind volumul de informatie care se poate stoca. Sistemul poate fi extins prin introducerea unui grup (cluster) de servere care sa mareasca cu mult posibilitatile de stocare. Pe ansamblu aplicatia realizata urmând modelul prezentat în acest articol s-a demonstrat a fi stabila, cu timpi de raspuns la interogari, mici chiar si în conditii de acces concurential sporit.

Bibliografie

1. Bryan Morgan, Jeff Perkins - *Working with Active ServerPages*, Macmillan Publishing USA, 1998
2. Jerry Honeycutt - *Special Edition Using HTML 4* Macmillan Publishing USA
3. Ryan K. Stephens, Ronald R Plew - *Sql Server in 21 Days*, Macmillan Publishing USA
4. Windows DNA XML Resource Kit – *Microsoft Publication, 2001*
5. *Sql Server OnLine Books(cd kit)*
6. Microsoft Developer Network (MSDN), *Microsoft Corporation*, Iulie 2001
7. Nikhil Kothari - *Introduction to Web Forms*, MsPress, Microsoft Corporation, Septembrie 2000