Evidenta articolelor primite la o revista

Ionica Madalina-Cristina, 332AB, Baze de date

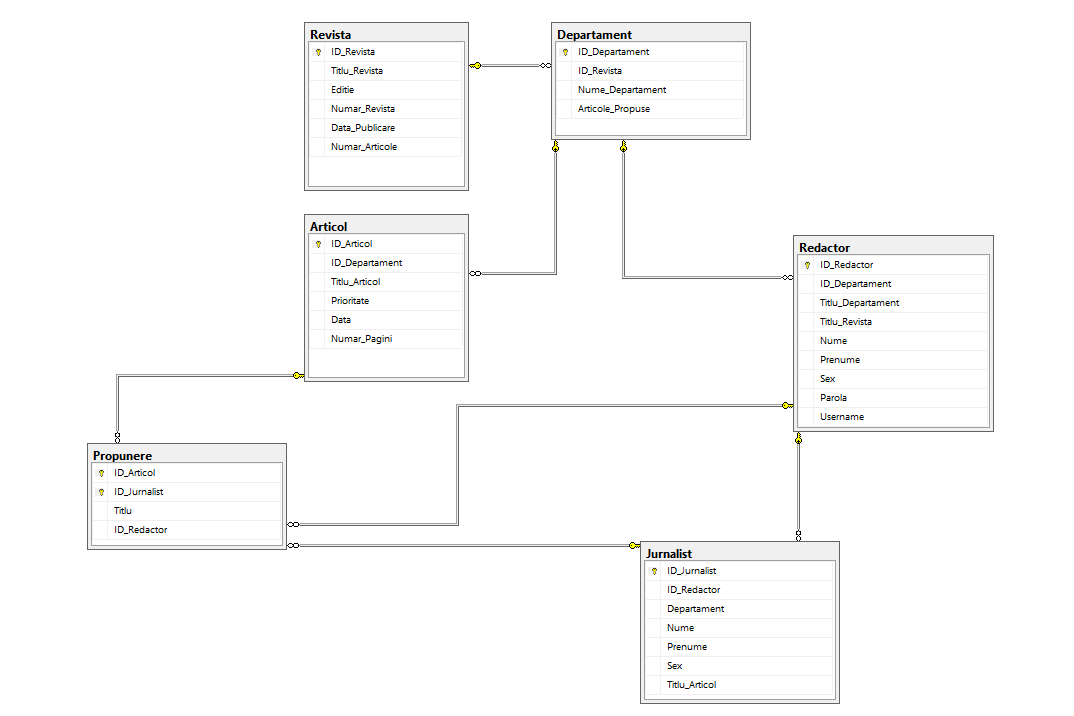
Procesul de publicare al unui articol din cadrul unei reviste presupune o serie de etape. Primul pas il presupune crearea si propunerea acestuia de catre un ournalist in cadrul departamentului din care face parte, in continuare articolul este analizat si astepta aprobarea redactorului de department care transmite mai departe spre publicare revistei, articolele.

In cadrul aplicatiei administratorul (redactorul) ar trebui sa:

* Faca modificari asupra datelor despre revista si departamentelor;
* Va putea centraliza numarul numarul de articole propuse, numarul de pagini al articolelor, dar si numarul departamentelor ascoiate fiecarei reviste
* Poate gestiona si supraveghea activitatea jurnalistilor aflalti in subordine
* Are posibilitatea de a verifica productivitatea lunara, dar si sa contorizese numarul de articole propuse spre publicare
* Poate vedea ce articole necesita revizurie imediata, tinand cont de prioritate

1. Definire relatii intre tabele
   * 1:1
     + ***Revista-Departament:*** O revista poate avea doar propriile departamente, iar un departament poate apartine doar unei reviste. Tabelul Revista reprezinta tabelul parinte al relatiei, iar tabelul Departament reprezinta tabelul copil.
   * 1:N
     + ***Departament-Redactor:*** In cadrul unei reviste pot exista mai multe departamente, dar fiecare departament are doar un redactor. Tabelul Redactor reprezinta partea „unu” a relatiei, iar tabelul Departament partea „mai multi”.
     + ***Departament-Articol:*** In cadrul unui departament pot fi create mai multe articole, dar un articol poate apartie doar unui singur departament. Tabelul Departament reprezinta partea „unu” a relatiei, iar tabelul Articol reprezinta partea „mai multi”.
     + ***Redactor-Propunere:*** Un redactor poate primi mai multe propuneri de publicare, dar o propunere de publicare nu poate fi trimisa la mai multi redactori. Tabelul Redactor reprezinta partea „unu” a relatiei, iar tabelul Propunere reprezinta partea „mai multi”.
     + ***Redactor-Jurnalist:*** Un redactor poate avea in subordine mai multi jurnalisti, dar un jurnalist poate fi coordonat doar de un redactor. Tabelul Redactor reprezinta partea „unu” a relatiei, iar tabelul Jurnalist reprezinta partea „mai multi”.
   * N:N
     + ***Propunere(Articol-Jurnalist):*** La un articol pot lucra mai multi jurnalisti, dar si un jurnalist poate propune mai multe articole spre aprobare, aceasta este o relatie many to many, de aceea vom introduce tabelul de legatura Propunere, unde vom avea doua relatii 1:N intre tabelele Articol-Propunere si Propunere-Jurnalist. Acesta va preluna copii ale cheilor primare ale celor doua tabelem devenind cheia primara compusa a tabelului.

In urma relatiilor descrise anterior am realizat diagrama:



Am construit tabelele alegand o cheie primara astfel incat sa indeplineasca proprietatile specifice ale acesteia fiind minimala, stabila si simpla, in plus, in functie de caz am definit constrangeri.

Tabel *Revista*

|  |
| --- |
| ID\_Revista CP (int)  Titlu\_Revista (nvarchar(50))  Editie (int)  Numar\_Revista (int)  Data\_Publicare (date)  Numar\_Articole (int) |

Tabel *Departament*

|  |
| --- |
| ID\_Departament CP (int)  ID\_Revista CE (int)  Nume\_Departament (nvarchar(80))  Articole\_Propuse (int) |

Tabel *Redactor*

|  |
| --- |
| ID\_Redactor CP (int)  ID\_Departament CE (int)  Titlu\_Departament (nvarchar(50))  Titlu\_Revista (nvarchar(50))  Nume (nvarchar(30))  Prenume (nvarchar(30))  Sex (char(1))  Parola (nvarchar(30))  Username (nvarchar(30)) |

*Constrangeri:* ([Sex]='F' OR [Sex]='M'), default 'F'

Tabel *Jurnalist*

|  |
| --- |
| ID\_Jurnalist CP (int)  ID\_Redactor CE (int)  Departament (nvarchar(50))  Nume (nvarchar(30))  Prenume (nvarchar(30))  Sex (char(1))  Titlu\_Articol (nvarchar(30)) |

*Constrangeri:* ([Sex]='F' OR [Sex]='M'), default 'F'

Tabel *Articol*

|  |
| --- |
| ID\_Articol CP (int)  ID\_Departament CE (int)  Titlu\_Articol (nvarchar(80))  Prioritate (int)  Data (date)  Numar\_Pagini (int) |

*Constrangeri:* ([Prioritate]='1' OR [Prioritate]='2' OR [Prioritate]='3' OR [Prioritate]='4' OR [Prioritate]='5'), default '1'

Tabel *Propunere*

create table Propunere (ID\_Articol int not null, ID\_Jurnalist int not null, Titlu nvarchar(15) not null, ID\_Redactor int not null);

ALTER TABLE Propunere

ADD primary key (ID\_Articol, ID\_Jurnalist)

Pentru a crea aplicatia vizuala am folosit C#, cu ajutorul ide-ului Visual Studio. In cadrul aplicatiei am afisat mai multe interogari.

***Interogari simple***

--1.Cate articole trimise redactorului apartin departamentului x?--

SELECT D.Nume\_Departament, COUNT(A.ID\_Articol) AS Numar\_Articole

FROM Departament D INNER JOIN Articol A ON D.ID\_Departament=A.ID\_Departament

WHERE D.Nume\_Departament='Fotbal'

GROUP BY D.Nume\_Departament

--2.Redactorii cu cei mai mult articole trimise de jurnalistii in subordine.--

SELECT R.Nume, R.Prenume, COUNT(J.ID\_Jurnalist) AS Articole\_Jurnalisti\_Subordine

FROM Redactor R INNER JOIN Jurnalist J ON R.ID\_Redactor=J.ID\_Redactor

GROUP BY R.Nume, R.Prenume

ORDER BY R.Nume, R.Prenume DESC

--3.Care sunt jurnalistii care au scris aricole de prioritate X? (prioritate intre 1 si 5)--

SELECT J.Nume, J.Prenume, J.Departament, A.Titlu\_Articol, A.Prioritate AS Prioritate\_Articole

FROM Articol A INNER JOIN Jurnalist J ON A.Titlu\_Articol=J.Titlu\_Articol

WHERE A.Prioritate=1

GROUP BY J.Nume, J.Prenume, J.Departament, A.Titlu\_Articol, A.Prioritate

--4.Cate articole are revista X?--

SELECT R.Titlu\_Revista, COUNT (A.ID\_Articol) AS Numar\_Articole

FROM Revista R INNER JOIN Departament D ON R.ID\_Revista=D.ID\_Revista INNER JOIN Articol A ON A.ID\_Departament=D.ID\_Departament

WHERE R.Titlu\_Revista='Vague'

GROUP BY R.Titlu\_Revista

--5.Cate pagini are fiecare revista?--

SELECT R.Titlu\_Revista, SUM(A.Numar\_Pagini) AS Numar\_Pagini

FROM Revista R INNER JOIN Departament D ON R.ID\_Revista=D.ID\_Revista INNER JOIN Articol A ON A.ID\_Departament=D.ID\_Departament

GROUP BY R.Titlu\_Revista

--6.Top 10 articole prioritare--

SELECT TOP 10 J.Nume, J.Prenume, J.Departament, A.Titlu\_Articol, COUNT(A.ID\_Articol) AS Prioritate\_Articol

FROM Jurnalist J INNER JOIN Articol A ON A.Titlu\_Articol=J.Titlu\_Articol

GROUP BY J.Nume, J.Prenume, J.Departament , A.Titlu\_Articol, A.Prioritate

ORDER BY A.Prioritate ASC

***Interogari complexe (cu subcereri)***

--1. Afisati titlul primelor 10 aricole in functie de numarul de pagini.--

SELECT TOP 10 A.Titlu\_Articol, A.Numar\_Pagini

FROM Articol A

WHERE A.Numar\_Pagini >= ANY (select A2.Numar\_Pagini

FROM Articol A2

GROUP BY A2.Numar\_Pagini )

GROUP BY A.Titlu\_Articol, A.Numar\_Pagini

ORDER BY A.Numar\_Pagini DESC

--2.In ce luna s-au scris cele mai multe articole?--

SELECT MONTH(A.Data) AS Luna\_Scriere, COUNT(A.ID\_Articol) AS Numar\_ArticoleScrise

FROM Articol A

GROUP BY MONTH(A.Data)

HAVING COUNT(A.Data) >= (SELECT TOP 1 count (A2.Data) FROM Articol A2 GROUP BY MONTH(A2.Data)

ORDER BY count(A2.Data) DESC)

--3.Numele si prenumele angajatilor care fac parte din departamentul x.--

SELECT DISTINCT J.Nume, J.Prenume

FROM Jurnalist J, (SELECT J2.ID\_Jurnalist

FROM Jurnalist J2

WHERE J2.Departament='Fotbal') AS J3

WHERE J.ID\_Jurnalist = J3.ID\_Jurnalist;

--4.Numele revistei si cate departamente are REVISTA-- AM PUS LA REVISTA

SELECT R.Titlu\_Revista,(

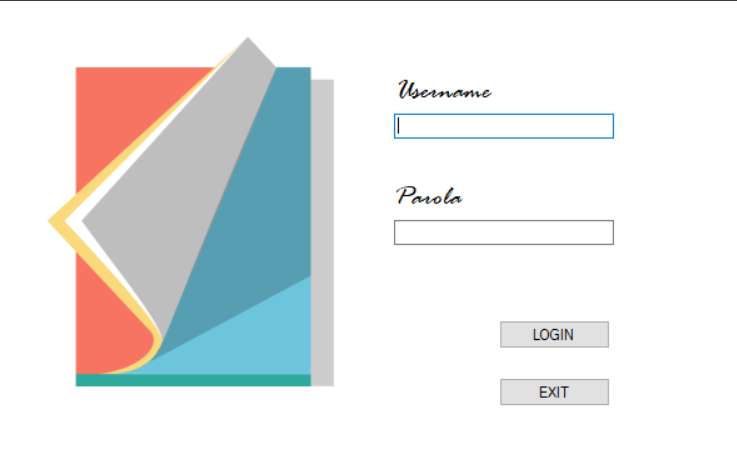
SELECT COUNT(\*) FROM Departament D

WHERE D.ID\_Revista=R.ID\_Revista)

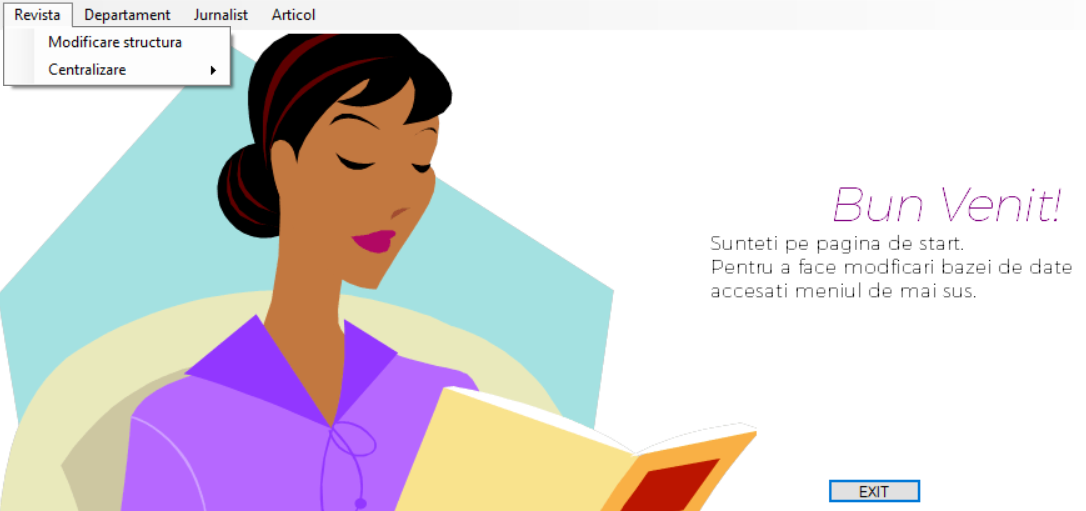
AS Revista\_Departament

FROM Revista R

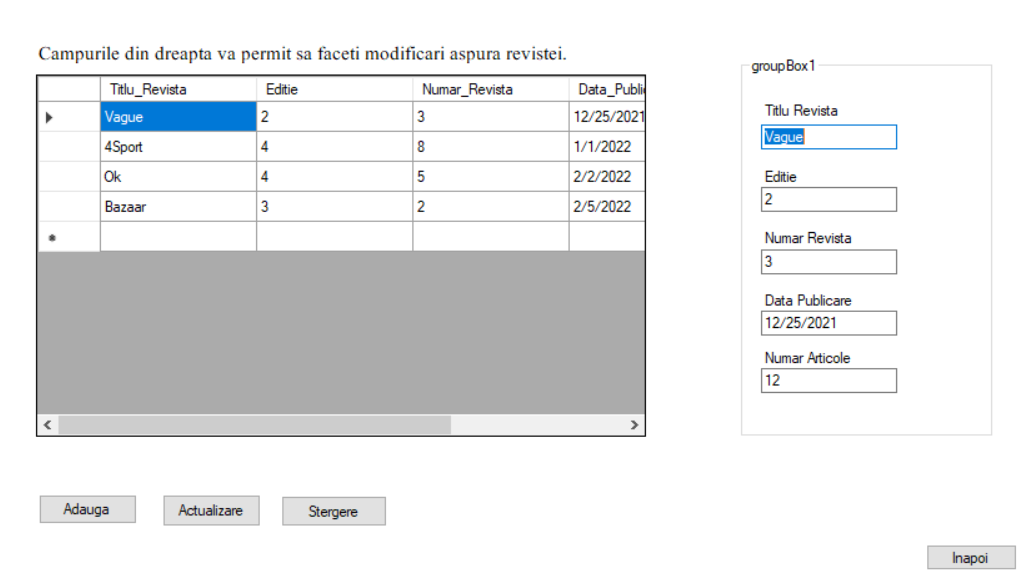
Aplicatia incepe cu o pagina de log in unde redactorul se poate conecta cu username-ul si parola.



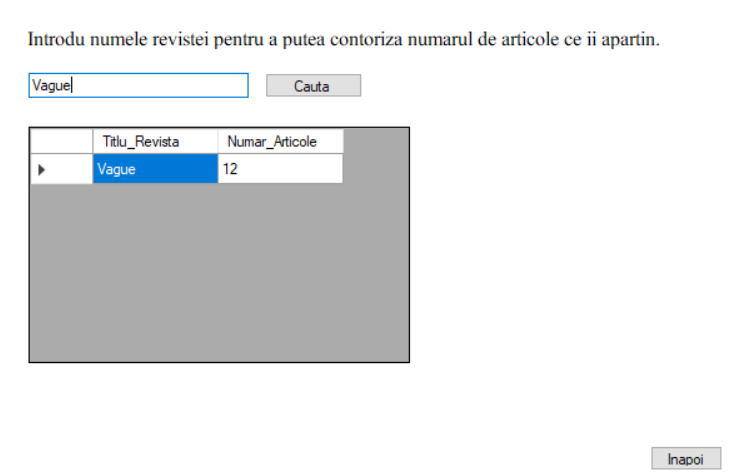
Daca logarea a fost facuta cu succes, se trece in zona de meniu:

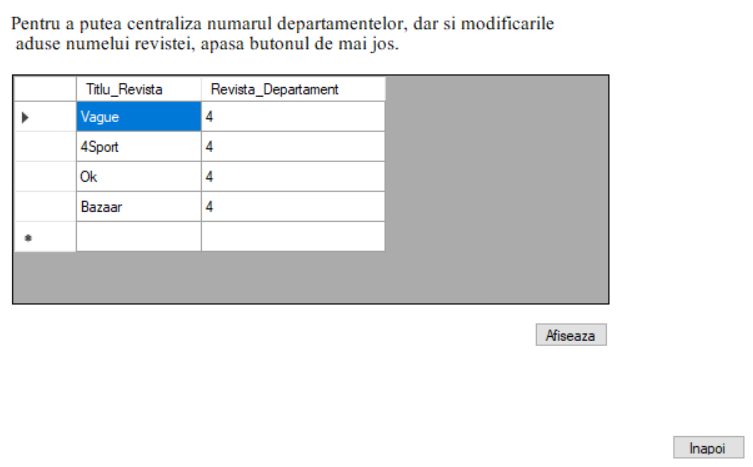


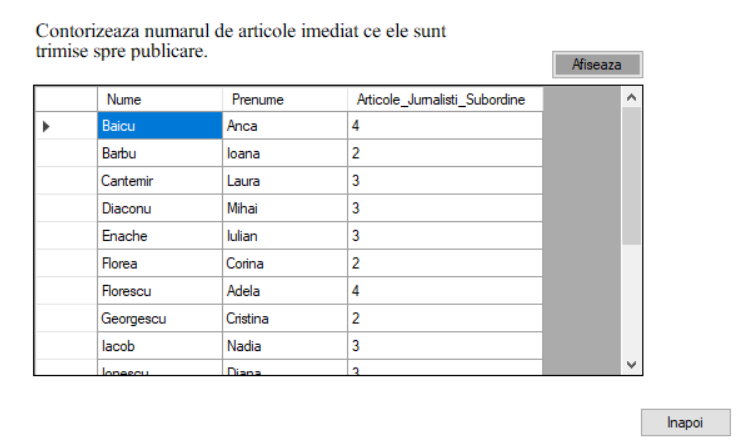
Din partea stanga sus se aleg optiunile. Pentru a modifica structura unui tabel:



Analog am creat si pentru tabelul Departament. Pentru interogari am avut o structura asemanatoare:







Pentru a face legatura intre interfata grafica si sql am folosit codul de mai jos caruia i-am schimbat doar definitia interogarii.

Exemplu:

try

{

using (SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["cn"].ConnectionString))

{

if (cn.State == ConnectionState.Open)

cn.Open();

using (DataTable dt = new DataTable("Revista"))

{

using (SqlCommand cmd = new SqlCommand("SELECT R.Titlu\_Revista, SUM(A.Numar\_Pagini) AS Numar\_Pagini FROM Revista R INNER JOIN Departament D ON R.ID\_Revista = D.ID\_Revista INNER JOIN Articol A ON A.ID\_Departament = D.ID\_Departament GROUP BY R.Titlu\_Revista ", cn))

{

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

adapter.Fill(dt);

dataGridView1.DataSource = dt;

}

}

}