

UNIVERSITATEA DIN BUCUREȘTI
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

PROIECT LA BAZE DE DATE

COORDONATOR ȘTIINȚIFIC:
VASILE SILVIU-LAURENȚIU

STUDENT:
FILIP RĂZVAN ADRIAN

BUCUREȘTI
2022

UNIVERSITATEA DIN BUCUREȘTI
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

REȚEA DE SOCIALIZARE

COORDONATOR ȘTIINȚIFIC:
VASILE SILVIU-LAURENȚIU

STUDENT:
FILIP RĂZVAN ADRIAN

BUCUREȘTI
2022

Cuprins

1.1. Prezentarea modelului de date	4
1.2. Condiții de funcționare.....	5
2.1. Diagramă Entitate-Relație	6
2.2. Entități	7
2.3. Relații	8
2.4. Atribute.....	9
2.4.1 Entitatea UTILIZATOR	9
2.4.2 Entitatea POSTĂRI	10
2.4.3 Entitatea PAGINI.....	10
2.4.4 Entitatea MESAJE_PRIVATE.....	10
2.4.5 Entitatea REGIUNI.....	11
2.4.6 Entitatea EVENIMENTE.....	11
2.4.7 Entitatea PIAȚĂ.....	11
3.1. Diagramă Conceptuală.....	14
3.2. Descrierea constrângerilor de integritate	15
3.2.1 Constrângeri de tip Primary key si Foreign key	15
3.2.2 Constrângeri de tip check.....	16
3.3. Schemele relaționale	17
3.4 Descrierea constrângerilor ON DELETE.....	17
4.1. Sintaxă implementare MySQL	19

1.1. Prezentarea modelului de date

În cele ce urmează voi prezenta modelul de date, restricțiile pe care trebuie să le respecte și voi afișa diagramele corespunzătoare.

Modelul de date va gestiona informații legate de organizarea și funcționarea unei rețele de socializare.

Rețelele de socializare sunt foarte utile în zilele noastre deoarece reprezintă un mod foarte rapid de a comunica cu alte persoane aflate la distanță de noi.

Punctul central al platformei îl reprezintă utilizatorii deoarece prin intermediul acestora este realizată dinamica paginii.

Utilizatorii pot trimite cereri de prietenie altor utilizatori, iar în funcție de răspunsul acestora (accept - 1 / reject - 0) se pot împrieteni pe platformă.

Mesajele private sunt destinate comunicării între utilizatori. Aceștia pot trimite mesaje oricui, indiferent dacă sunt sau nu prieteni pe platformă.

Paginile sunt dedicate persoanelor publice și au ca scop postarea informațiilor legate de persoana respectivă. Aceste pagini pot fi apreciate de către utilizatori , însa nu aparțin utilizatorilor.

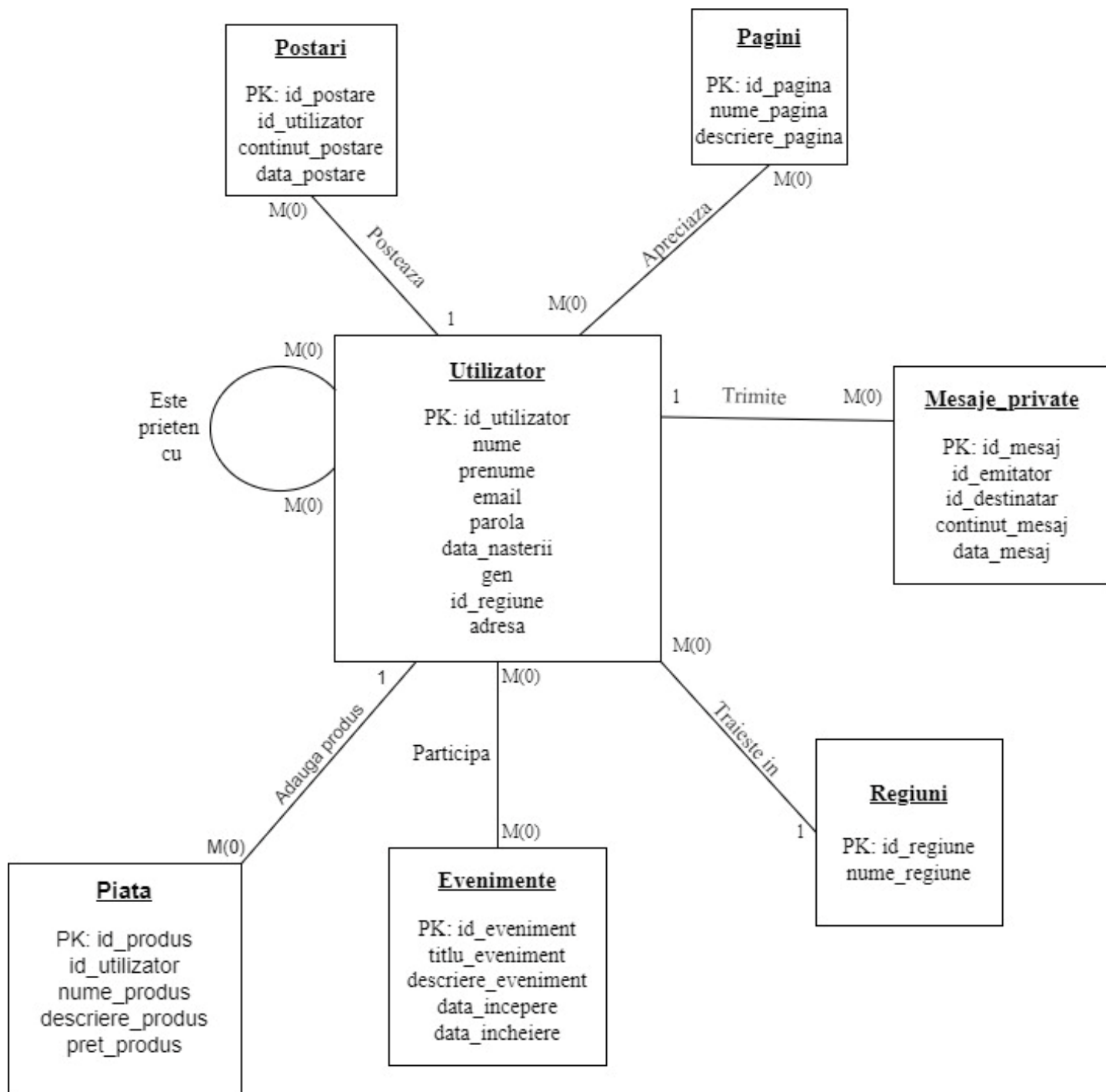
Pe platformă pot fi create diferite evenimente, acestea având o dată de începere și una de încheiere, iar utilizatorii se pot înscrie pentru a participa la acestea.

Piața reprezintă locul unde se pot posta diferite produse cu scopul de a le vinde. Utilizatorul își poate posta produsul/produsele, iar potențialii cumpărători îl pot contacta printr-un mesaj privat pentru mai multe detalii.

1.2. Condiții de funcționare

- O pagină poate fi apreciată de mai mulți utilizatori (sau niciunul). La rândul lor, utilizatorii pot aprecia mai multe pagini (sau niciuna).
- Un utilizator poate face mai multe postări (sau niciuna), dar o postare poate fi făcută de un singur utilizator.
- Un mesaj poate fi trimis de un singur utilizator, dar un utilizator poate trimite mai multe mesaje (sau niciunul).
- Un utilizator poate participa la mai multe evenimente (sau la niciunul), iar la un eveniment pot participa mai mulți utilizatori - pot exista evenimente fără participanți.
- Utilizatorul poate face parte dintr-o singură regiune - pot exista regiuni din care nu face parte niciun utilizator.
- În secțiunea “Piață”, un utilizator poate posta mai multe produse (sau niciun produs).

2.1. Diagramă Entitate-Relatie



2.2. Entități

Pentru modelul de date referitor la rețeaua de socializare, structurile *UTILIZATOR*, *POSTĂRI*, *PAGINI*, *MESAJE_PRIVATE*, *REGIUNI*, *EVENIMENTE*, *PIAȚĂ* reprezintă entități. Voi prezenta entitățile modelului de date, dând o descriere completă a fiecăreia. De asemenea, pentru fiecare entitate se va preciza cheia primară.

UTILIZATOR = persoană care interacționează cu alte persoane prin intermediul platformei. Cheia primară a entității este *id_utilizator*.

POSTĂRI = postări pe care un utilizator le poate face pe platforma. Cheia primară a entității este *id_postare*.

PAGINI = pagini destinate persoanelor publice ce conțin informații legate de acestea. *Id_pagină* este cheia primară a acestei entități.

MESAJE_PRIVATE = mesajele private trimise între doi utilizatori. Cheia primară a entității este *id_mesaj*.

REGIUNI = lista regiunilor din care utilizatorii pot face parte. *Id_regiune* reprezintă cheia primară a acestei entități.

EVENIMENTE = lista de evenimente de pe platformă, având ca și cheie primară *id_eveniment*.

PIAȚĂ = locul unde utilizatorii pot posta diferite lucruri pentru a le vinde. Cheia primară a entității este *id_produs*.

2.3. Relații

În cele ce urmează voi prezenta relațiile modelului de date, dând o descriere amplă a fiecăreia. De asemenea, denumirile acestor legături sunt sugestive, reflectând conținutul acestora și entitățile pe care le leagă.

Pentru fiecare relație voi preciza cardinalitatea minimă și maximă.

UTILIZATOR – postează - POSTĂRI = relație care leagă entitățile utilizator și postări, reflectând legătura dintre acestea (ce postări a făcut un utilizator). Ea are cardinalitatea minimă 1:0 și cea maximă 1:n.

UTILIZATOR – apreciază – PAGINI = relație care leagă entitățile utilizator și PAGINI, reflectând legătura dintre acestea (ce pagini apreciază un utilizator). Relația este de tip many-to-many, cardinalitatea minimă fiind 0:0, iar cea maximă fiind m:n.

UTILIZATOR – trimite – MESAJE_PRIVATE = relație care leagă entitățile utilizator și mesaje_private, reflectând legătura dintre acestea (ce mesaje private a trimis un utilizator). Ea are cardinalitatea minimă 1:0 și cea maximă 1:n.

UTILIZATOR – trăiește în - REGIUNI = relație care leagă entitățile utilizator și regiuni, reflectând legătura dintre acestea (în ce regiune locuiește un utilizator). Ea are cardinalitatea minimă 0:1 și cardinalitatea maximă n:1.

UTILIZATOR – participă - EVENIMENTE = relație care leagă entitățile utilizator și evenimente, reflectând legătura dintre acestea (la ce evenimente participă un utilizator). Relația este de tip many-to-many, cardinalitatea minimă fiind 0:0, iar cea maximă fiind m:n.

UTILIZATOR – adaugă produs - PIAȚĂ = relație care leagă entitățile utilizator și piață, reflectând legătura dintre acestea (ce produse are un utilizator postate spre vânzare). Ea are cardinalitatea minimă 1:0 și cardinalitatea maximă 1:n.

UTILIZATOR - este prieten cu - UTILIZATOR = relație care leagă entitatea utilizator de ea însăși (reflectă prietenii unui utilizator). Relația este de tip many-to-many, cardinalitatea minimă fiind 0:0, iar cea maximă fiind m:n.

2.4. Atribute

2.4.1 Entitatea UTILIZATOR are ca atribute:

- id_utilizator = variabilă de tip întreg, de lungime maximă 10, care reprezintă codul unui utilizator.
- nume = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele unui utilizator.
- prenume = variabilă de tip caracter, de lungime maximă 30, care reprezintă prenumele unui utilizator.
- email = variabilă de tip caracter, de lungime maximă 30, care reprezintă adresa de email a unui utilizator.
- parolă = variabilă de tip caracter, de lungime maximă 30, care reprezintă parola unui utilizator.
- data_nașterii = variabilă de tip dată calendaristică ce reprezintă ziua de naștere a utilizatorului.
- gen = variabilă de tip caracter, de lungime 1, care poate lua valorile b sau f în funcție de sexul utilizatorului.
- id_regiune = variabilă de tip întreg, de lungime maximă 1, ce reprezintă codul regiunii în care locuiește utilizatorul. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul REGIUNI.
- adresă = variabilă de tip întreg, de lungime maximă 30, care reprezintă locația (orașul) în care utilizatorul locuiește.

2.4.2 Entitatea POSTĂRI are ca attribute:

- id_postare = variabilă de tip întreg, de lungime maximă 20, care reprezintă codul postării.
- id_utilizator = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care a făcut postarea. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul UTILIZATOR.
- conținut_postare = variabilă de tip caracter, de lungime maximă 300, care reprezintă conținutul postării.
- dată_postare = variabilă de tip dată calendaristică ce reprezintă data la care a fost făcută postarea.

2.4.3 Entitatea PAGINI are ca attribute:

- id_pagină = variabilă de tip întreg, de lungime maximă 10, care reprezintă codul paginii.
- nume_pagină = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele paginii.
- descriere_pagină = variabilă de tip caracter, de lungime maximă 300, care reprezintă descrierea paginii.

2.4.4 Entitatea MESAJE_PRIVATE are ca attribute:

- id_mesaj = variabilă de tip întreg, de lungime maximă 20, care reprezintă codul mesajului.
- id_emițător = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care a trimis mesajul. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul UTILIZATOR.
- id_destinatar = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care a primit mesajul. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul UTILIZATOR.
- conținut_mesaj = variabilă de tip caracter, de lungime maximă 300, care reprezintă conținutul mesajului.

- `dată_mesaj` = variabilă de tip dată calendaristică ce reprezintă data la care a fost trimis mesajul.

2.4.5 Entitatea REGIUNI are ca atribute:

- `id_regiune` = variabilă de tip întreg, de lungime maximă 1, care reprezintă codul regiunii.

- `nume_regiune` = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele regiunii.

2.4.6 Entitatea EVENIMENTE are ca atribute:

- `id_eveniment` = variabilă de tip întreg, de lungime maximă 10, care reprezintă codul evenimentului.

- `titlu_eveniment` = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele evenimentului.

- `descriere_eveniment` = variabilă de tip caracter, de lungime maximă 300, care reprezintă descrierea evenimentului.

- `dată_incepere` = variabilă de tip dată calendaristică ce reprezintă data de începere a evenimentului.

- `dată_incheiere` = variabilă de tip dată calendaristică ce reprezintă data de încheiere a evenimentului.

2.4.7 Entitatea PIAȚĂ are ca atribute:

- `id_produs` = variabilă de tip întreg, de lungime maximă 10, care reprezintă codul produsului listat pe piață.

- `id_utilizator` = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care a postat produsul. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul UTILIZATOR.

- `nume_produs` = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele produsului.

- descriere_produs = variabilă de tip caracter, de lungime maximă 300, care reprezintă descrierea produsului.

- preț_produs = variabilă de tip întreg, de lungime maximă 10, care reprezintă prețul produsului.

Relația UTILIZATOR – apreciază - PAGINI are ca attribute:

- id_pagină = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul paginii. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul PAGINI.

- id_utilizator = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care a apreciat pagina. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul UTILIZATOR.

Relația UTILIZATOR – participă - EVENIMENT are ca attribute:

- id_utilizator = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care participă la eveniment. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul UTILIZATOR.

- id_eveniment = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul evenimentului. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul EVENIMENTE.

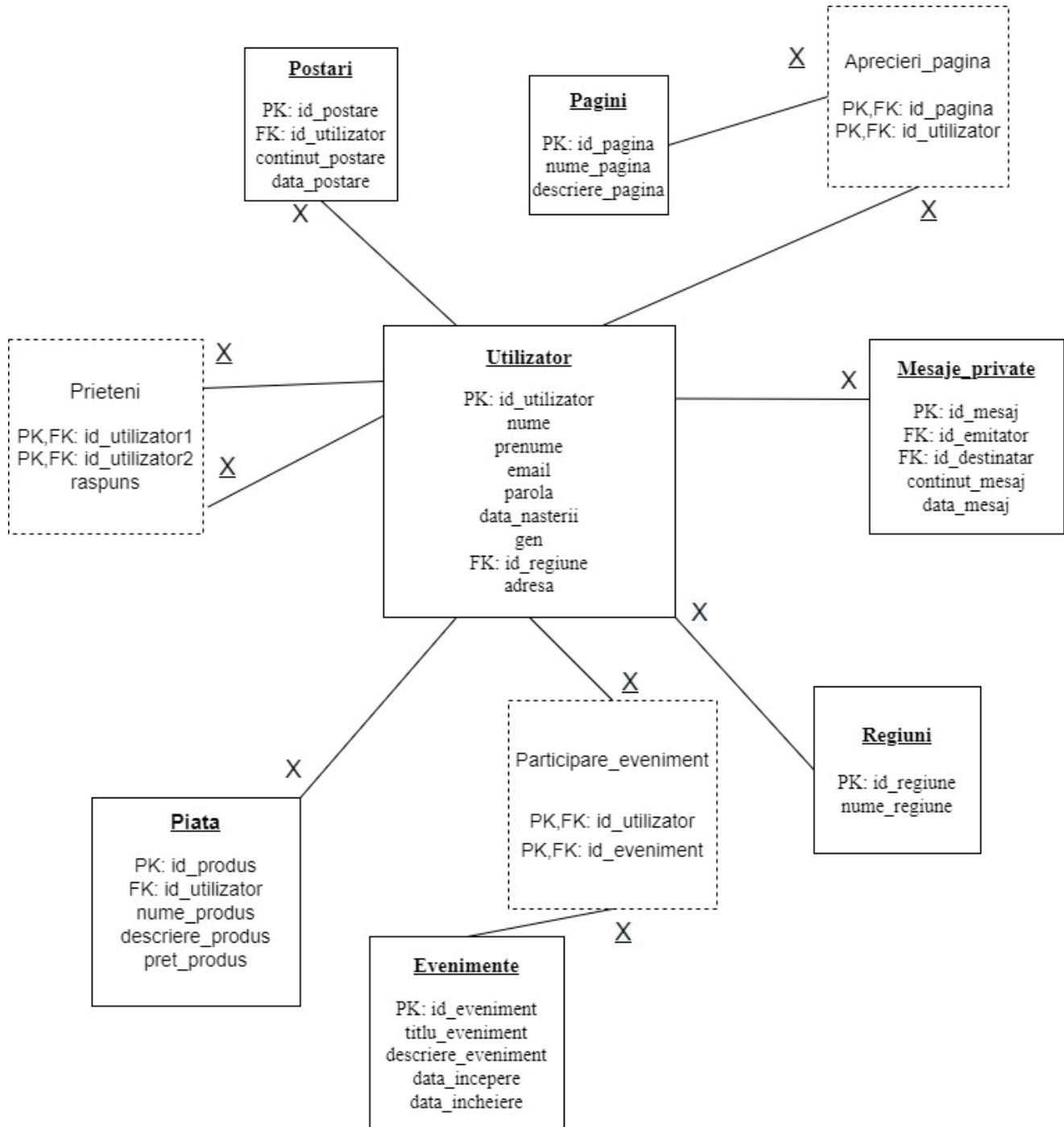
Relația UTILIZATOR – este prieten cu - UTILIZATOR are ca attribute:

- id_utilizator1 = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care este prieten cu id_utilizator2. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul UTILIZATOR.

- id_utilizator2 = variabilă de tip întreg, de lungime maximă 10, ce reprezintă codul utilizatorului care este prieten cu id_utilizator1. Atributul trebuie să corespundă cu o valoare a cheii primare din tabelul UTILIZATOR.

- răspuns = variabilă de tip întreg, de lungime maximă 1, având ca valori 0 sau 1 în funcție de răspunsul dat de utilizatorul care primește cererea de prietenie (0 = cerere neacceptată / 1 = cerere acceptată).

3.1. Diagramă Conceptuală



3.2. Descrierea constrângerilor de integritate

3.2.1 Constrângeri de tip Primary key si Foreign key

UTILIZATOR -> Cheie primară: *id_utilizator*

-> Cheie externă: *id_regiune*, referă coloana *id_regiune* din tabela REGIUNI

POSTĂRI -> Cheie primară: *id_postare*

-> Cheie externă: *id_utilizator*, referă coloana *id_utilizator* din tabela UTILIZATOR

PAGINI -> Cheie primară: *id_pagină*

MESAJE_PRIVATE -> Cheie primară: *id_mesaj*

-> Cheie externă: *id_emitator*, referă coloana *id_utilizator* din tabela UTILIZATOR

-> Cheie externă: *id_destinatar*, referă coloana *id_utilizator* din tabela UTILIZATOR

REGIUNI -> Cheie primară: *id_regiune*

EVENIMENTE -> Cheie primară: *id_eveniment*

PIAȚĂ -> Cheie primară: *id_produc*

-> Cheie externă: *id_utilizator*, referă coloana *id_utilizator* din tabela UTILIZATOR

PRIETENI -> Cheie primară compusă: id_utilizator , id_utilizator2

-> Cheie externă: id_utilizator1, referă coloana id_utilizator din tabela UTILIZATOR

-> Cheie externă: id_utilizator2, referă coloana id_utilizator din tabela UTILIZATOR

APRECIERI_PAGINĂ -> Cheie primară compusă: id_pagina , id_utilizator

-> Cheie externă: id_pagină, referă coloana id_pagină din tabela PAGINI

-> Cheie externă: id_utilizator, referă coloana id_utilizator din tabela UTILIZATOR

PARTICIPARE_EVENTIMENT -> Cheie primară compusă: id_utilizator , id_eventiment

-> Cheie externă: id_utilizator, referă coloana id_utilizator din tabela UTILIZATOR

-> Cheie externă: id_eventiment, referă coloana id_eventiment din tabela EVENIMENTE

3.2.2 Constrângeri de tip check

“CHECK (gen in ('b','f')) “ -> Verifică dacă gen-ul utilizatorului este bărbat (b) sau femeie (f).

3.3. Schemele relaționale

- UTILIZATOR(id_utilizator#, nume, prenume, email, parolă, data_nașterii, gen, id_regiune, adresă);
- POSTĂRI(id_postare#, id_utilizator, conținut_postare, dată_postare);
- PAGINI(id_pagină#, nume_pagină, descriere_pagină);
- APRECIERI_PAGINĂ(id_pagină#, id_utilizator#);
- MESAJE_PRIVATE (id_mesaj#, id_emițător, id_destinatar, conținut_mesaj, dată_mesaj);
- REGIUNI (id_regiune#, nume_regiune);
- PARTICIPARE_EVENT (id_utilizator#, id_event#);
- EVENTE (id_event#, titlu_event, descriere_event, dată_începere, dată_încheiere);
- PIAȚĂ (id_produs#, id_utilizator, nume_produs, descriere_produs, preț_produs);
- PRIETENI(id_utilizator1#, id_utilizator2#, răspuns);

3.4 Descrierea constrângerilor ON DELETE

În model sunt prezente 11 constrângeri de tip ON DELETE.

- `gen_constraint` - constrângere de tip „on delete set null”.
Dacă o regiune este ștearsă, utilizatorului i se atribuie valoarea NULL în atributul “id_regiune” din tabela UTILIZATOR, în cazul în care acesta avea atribuită acea regiune.
- `utilizator_piață_fk` - constrângere de tip „on delete cascade”.
Dacă un utilizator este ștears, și produsul listat de acesta spre vânzare va fi ștears.

- `utilizator_destinatar_fk` - constrângere de tip „on delete cascade”. Dacă un utilizator este șters, vor fi șterse, de asemenea, și mesajele private pe care acesta le-a primit.
- `utilizator_emitator_fk` - constrângere de tip „on delete cascade”. Dacă un utilizator este șters, vor fi șterse, de asemenea, și mesajele private pe care acesta le-a trimis.
- `utilizator_postare_fk` - constrângere de tip „on delete cascade”. Dacă un utilizator este șters, vor fi șterse și postările acestuia.
- `utilizator1_prieteni_fk` , `utilizator2_prieteni_fk` - constrângeri de tip „on delete cascade”. Dacă un utilizator este șters, vor fi ștersi și prietenii acestuia.
- `eveniment_evenpart_fk` - constrângere de tip „on delete cascade”. Dacă un eveniment este șters, vor fi șterse și înscrierile utilizatorilor la acel eveniment.
- `utilizator_eveniment_fk` - constrângere de tip „on delete cascade”. Dacă un utilizator este șters, vor fi șterse și participările acestuia la evenimentele la care acesta s-a înscris.
- `aprecieri_pagina_fk` - constrângere de tip „on delete cascade”. Dacă o pagină este ștearsă, vor fi șterse și aprecierile respectivei pagini.
- `utilizator_aprecieri_fk` - constrângere de tip „on delete cascade”. Dacă un utilizator este șters, vor fi șterse și aprecierile acestuia față de anumite pagini.

4.1. Sintaxă implementare MySQL

```
CREATE TABLE `regiuni` (  
  `id_regiune` int(11) NOT NULL AUTO_INCREMENT,  
  `nume_regiune` varchar(30) NOT NULL,  
  PRIMARY KEY (`id_regiune`)  
)
```

```
CREATE TABLE `utilizator` (  
  `id_utilizator` int(10) NOT NULL AUTO_INCREMENT,  
  `nume` varchar(30) NOT NULL,  
  `prenume` varchar(30) NOT NULL,  
  `email` varchar(30) NOT NULL,  
  `parola` varchar(30) NOT NULL,  
  `data_nasterii` date NOT NULL,  
  `gen` varchar(1) NOT NULL CHECK (`gen` in ('b','f')),  
  `id_regiune` int(1) DEFAULT NULL,  
  `adresa` varchar(30) NOT NULL,  
  PRIMARY KEY (`id_utilizator`),  
  KEY `gen_constraint` (`id_regiune`),  
  CONSTRAINT `gen_constraint` FOREIGN KEY (`id_regiune`) REFERENCES `regiuni`  
  (`id_regiune`) ON DELETE SET NULL  
)
```

```
CREATE TABLE `pagini` (  
  `id_pagina` int(10) NOT NULL AUTO_INCREMENT,  
  `nume_pagina` varchar(30) NOT NULL,  
  `descriere_pagina` varchar(300) NOT NULL,  
  PRIMARY KEY (`id_pagina`)
```

)

```
CREATE TABLE `evenimente` (  
  `id_eveniment` int(10) NOT NULL AUTO_INCREMENT,  
  `titlu_eveniment` varchar(30) NOT NULL,  
  `descriere_eveniment` varchar(300) NOT NULL,  
  `data_incepere` date NOT NULL,  
  `data_incheiere` date NOT NULL,  
  PRIMARY KEY (`id_eveniment`)  
)
```

```
CREATE TABLE `piata` (  
  `id_produs` int(10) NOT NULL AUTO_INCREMENT,  
  `id_utilizator` int(10) NOT NULL,  
  `nume_produs` varchar(30) NOT NULL,  
  `descriere_produs` varchar(300) DEFAULT NULL,  
  `pret_produs` int(10) NOT NULL,  
  PRIMARY KEY (`id_produs`),  
  KEY `utilizator_piata_fk` (`id_utilizator`),  
  CONSTRAINT `utilizator_piata_fk` FOREIGN KEY (`id_utilizator`) REFERENCES  
  `utilizator` (`id_utilizator`) ON DELETE CASCADE  
)
```

```
CREATE TABLE `mesaje_private` (  
  `id_mesaj` int(20) NOT NULL AUTO_INCREMENT,  
  `id_emitator` int(10) NOT NULL,  
  `id_destinatar` int(10) NOT NULL,  
  `continut_mesaj` varchar(300) NOT NULL,  
  `data_mesaj` date NOT NULL,
```

```

PRIMARY KEY (`id_mesaj`),
KEY `utilizator_emitator_fk` (`id_emitator`),
KEY `utilizator_destinatar_fk` (`id_destinatar`),
CONSTRAINT `utilizator_destinatar_fk` FOREIGN KEY (`id_destinatar`) REFERENCES
`utilizator` (`id_utilizator`) ON DELETE CASCADE ,
CONSTRAINT `utilizator_emitator_fk` FOREIGN KEY (`id_emitator`) REFERENCES
`utilizator` (`id_utilizator`) ON DELETE CASCADE
)

```

```

CREATE TABLE `postari` (
  `id_postare` int(20) NOT NULL AUTO_INCREMENT,
  `id_utilizator` int(10) NOT NULL,
  `continut_postare` varchar(300) NOT NULL,
  `data_postare` date NOT NULL,
  PRIMARY KEY (`id_postare`),
  KEY `utilizator_postare_fk` (`id_utilizator`),
  CONSTRAINT `utilizator_postare_fk` FOREIGN KEY (`id_utilizator`) REFERENCES
`utilizator` (`id_utilizator`) ON DELETE CASCADE
)

```

----- Tabele asociative -----

```
CREATE TABLE `prietenii` (  
  `id_utilizator1` int(10) NOT NULL,  
  `id_utilizator2` int(10) NOT NULL,  
  `raspuns` int(11) NOT NULL,  
  PRIMARY KEY (`id_utilizator1`,`id_utilizator2`),  
  KEY `utilizator2_prietenii_fk` (`id_utilizator2`),  
  CONSTRAINT `utilizator1_prietenii_fk` FOREIGN KEY (`id_utilizator1`) REFERENCES  
`utilizator` (`id_utilizator`) ON DELETE CASCADE,  
  CONSTRAINT `utilizator2_prietenii_fk` FOREIGN KEY (`id_utilizator2`) REFERENCES  
`utilizator` (`id_utilizator`) ON DELETE CASCADE  
)
```

```
CREATE TABLE `participare_eveniment` (  
  `id_utilizator` int(10) NOT NULL,  
  `id_eveniment` int(10) NOT NULL,  
  PRIMARY KEY (`id_utilizator`,`id_eveniment`),  
  KEY `eveniment_evenpart_fk` (`id_eveniment`),  
  CONSTRAINT `eveniment_evenpart_fk` FOREIGN KEY (`id_eveniment`) REFERENCES  
`evenimente` (`id_eveniment`) ON DELETE CASCADE ,  
  CONSTRAINT `utilizator_eveniment_fk` FOREIGN KEY (`id_utilizator`) REFERENCES  
`utilizator` (`id_utilizator`) ON DELETE CASCADE  
)
```

```
CREATE TABLE `aprecieri_pagina` (  
  `id_pagina` int(10) NOT NULL,  
  `id_utilizator` int(10) NOT NULL,
```

```

PRIMARY KEY (`id_pagina`,`id_utilizator`),
KEY `utilizator_aprecieri_fk` (`id_utilizator`),
CONSTRAINT `aprecieri_pagina_fk` FOREIGN KEY (`id_pagina`) REFERENCES `pagini`
(`id_pagina`) ON DELETE CASCADE ,
CONSTRAINT `utilizator_aprecieri_fk` FOREIGN KEY (`id_utilizator`) REFERENCES
`utilizator` (`id_utilizator`) ON DELETE CASCADE
)

```

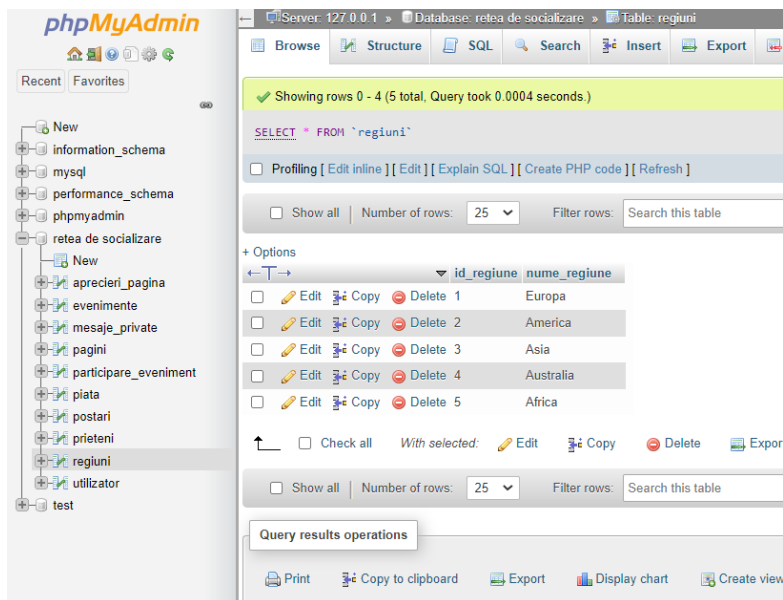
----- Inserare date -----

Inserare REGIUNI

```

INSERT INTO `regiuni` (`id_regiune`, `nume_regiune`) VALUES
(1, 'Europa'),
(2, 'America'),
(3, 'Asia'),
(4, 'Australia'),
(5, 'Africa');

```



Inserare UTILIZATOR

```
INSERT INTO `utilizator` (`id_utilizator`, `nume`, `prenume`, `email`, `parola`, `data_nasterii`,  
`gen`, `id_regiune`, `adresa`) VALUES
```

(1, 'Filip', 'Razvan', 'razvan.filip@s.unibuc.ro', 'razvan1234', '2001-01-15', 'b', 1, 'Razvad, str. 1 Ploiesti , nr.28'),

(2, 'Aciu', 'Mircea', 'aciu.mircea@s.unibuc.ro', 'mircea1234', '1995-02-18', 'b', 1, 'Targoviste, str. 1 Mai, nr.42'),

(3, 'Chelaru', 'Gabriela', 'gabriela.chelaru@s.unibuc.ro', 'gabriela1234', '2001-10-08', 'f', 1, 'Baleni, soseaua Bucuresti-Targ'),

(4, 'Cernat', 'Mihai', 'mihai.cernat@s.unibuc.ro', 'mihai1234', '2002-05-13', 'b', 1, 'Targoviste, str. 8 Martie, nr.2'),

(5, 'Adam', 'David', 'david.adam@gmail.com', 'david1234', '2005-04-11', 'b', 2, '2115 w Farwell ave Chicago il '),

(6, 'Tom', 'Oliver', 'oliver.tom@gmail.com', 'oliver1234', '1993-09-22', 'b', 2, 'Northwest Arctic Ambler zip co'),

(7, 'Ashley', 'Adele', 'adele.ashley@gmail.com', 'adele1234', '1994-03-12', 'f', 2, 'Northwest Arctic Deering zip c'),

(8, 'Darie', 'Eleonora', 'eleonora.darie@gmail.com', 'eleonora1234', '1990-04-18', 'f', 1, 'Targoviste, str. Cetatii nr.1'),

(9, 'Miclaus', 'Simona', 'simona.miclaus@yahoo.com', 'simona1234', '2000-09-22', 'f', 1, 'Teis, str. Ialomitei nr.43'),

(10, 'Neamt', 'Liviu', 'liviu.neamt@yahoo.com', 'liviu1234', '2002-10-25', 'b', 1, 'Gorgota, str. Macilor nr. 22');

The screenshot displays the phpMyAdmin interface for a MySQL database named 'retea de socializare'. The 'utilizator' table is selected, showing 10 rows of data. The table structure includes columns for id_utilizator, nume, prenume, email, parola, data_nasterii, gen, id_regiune, and adresa. The data rows correspond to the SQL insert statements provided in the document.

	id_utilizator	nume	prenume	email	parola	data_nasterii	gen	id_regiune	adresa
<input type="checkbox"/>	1	Filip	Razvan	razvan.filip@s.unibuc.ro	razvan1234	2001-01-15	b	1	Razvad, str. 1 Ploiesti , nr.28
<input type="checkbox"/>	2	Aciu	Mircea	aciu.mircea@s.unibuc.ro	mircea1234	1995-02-18	b	1	Targoviste, str. 1 Mai, nr.42
<input type="checkbox"/>	3	Chelaru	Gabriela	gabriela.chelaru@s.unibuc.ro	gabriela1234	2001-10-08	f	1	Baleni, soseaua Bucuresti-Targ
<input type="checkbox"/>	4	Cernat	Mihai	mihai.cernat@s.unibuc.ro	mihai1234	2002-05-13	b	1	Targoviste, str. 8 Martie, nr.2
<input type="checkbox"/>	5	Adam	David	david.adam@gmail.com	david1234	2005-04-11	b	2	2115 w Farwell ave Chicago il
<input type="checkbox"/>	6	Tom	Oliver	oliver.tom@gmail.com	oliver1234	1993-09-22	b	2	Northwest Arctic Ambler zip co
<input type="checkbox"/>	7	Ashley	Adele	adele.ashley@gmail.com	adele1234	1994-03-12	f	2	Northwest Arctic Deering zip c
<input type="checkbox"/>	8	Darie	Eleonora	eleonora.darie@gmail.com	eleonora1234	1990-04-18	f	1	Targoviste, str. Cetatii nr.1
<input type="checkbox"/>	9	Miclaus	Simona	simona.miclaus@yahoo.com	simona1234	2000-09-22	f	1	Teis, str Ialomitei nr.43
<input type="checkbox"/>	10	Neamt	Liviu	liviu.neamt@yahoo.com	liviu1234	2002-10-25	b	1	Gorgota, str. Macilor nr. 22

Inserare PAGINI

INSERT INTO `pagini` (`id_pagina`, `nume_pagina`, `descriere_pagina`) VALUES

(1, 'Mihai Bendeac', 'Mihai Bendeac este un actor de teatru și film, jucând în filmele Milionari de Weekend și Supraviețuitorul. Mihai a scris toate cele 6 sezoane În Puii Mei pe care le-a și interpretat. A mai jucat în show-urile La Bloc, Românii au Artiști, Băieți de Oraș, iComedy, Mondenii, Jurnalul unui burlac și în '),

(2, 'Delia Matache', 'Delia Matache, cunoscută mai bine sub numele de scenă Delia, este o cântăreață, compozitoare, dansatoare, vedetă de televiziune și jurată în cadrul competiției muzicale internaționale X Factor România și a emisiunii iUmor.'),

(3, 'Cătălin Ștefan Ion', 'Cătălin Ștefan Ion, cunoscut sub pseudonimul Cheloo, este un rapper român, unul din membrii formației române de hip-hop Paraziții. A lansat primul său album de studio solo intitulat Sindromul Tourette în anul 2003, album premiat cu Discul de Aur pentru numărul mare de exemplare vândute.'),

(4, 'Andreea Raicu', 'Andreea Raicu este o vedetă de televiziune din România. A debutat ca model, câștigând câteva importante concursuri de modelling apoi a intrat în televiziune unde s-a consacrat prezentând emisiuni importante de divertisment ca Big Brother și Megastar.'),

(5, 'Laura Petrescu', 'Laura Petrescu , cunoscută sub numele de scenă Lora, este o cântăreață de muzică pop-dance, actriță, fotomodel, prezentatoare a emisiunii \"Vorbește lumea\", de la Pro Tv și vedetă de televiziune din România. Înainte de a-și începe cariera solo, a făcut parte din formația Wassabi. S-a născut în Vaslui');

The screenshot shows the phpMyAdmin interface. On the left is a sidebar with a tree view of databases and tables. The main area displays the 'pagini' table from the 'retea de socializare' database. The table has three columns: 'id_pagina', 'nume_pagina', and 'descriere_pagina'. It contains 5 rows of data. Below the table, there are options to check all rows, edit, copy, delete, or export the selected data. At the bottom, there is a section for 'Query results operations' with buttons for print, copy to clipboard, export, display chart, and create view. A 'Bookmark this SQL query' dialog is also visible.

	id_pagina	nume_pagina	descriere_pagina
<input type="checkbox"/>	1	Mihai Bendeac	Mihai Bendeac este un actor de teatru și film, juc...
<input type="checkbox"/>	2	Delia Matache	Delia Matache, cunoscută mai bine sub numele de sc...
<input type="checkbox"/>	3	Cătălin Ștefan Ion	Cătălin Ștefan Ion, cunoscut sub pseudonimul Chelo...
<input type="checkbox"/>	4	Andreea Raicu	Andreea Raicu este o vedetă de televiziune din Rom...
<input type="checkbox"/>	5	Laura Petrescu	Laura Petrescu , cunoscută sub numele de scenă Lor...

Inserare POSTARI

```
INSERT INTO `postari` (`id_postare`, `id_utilizator`, `continut_postare`, `data_postare`) VALUES
```

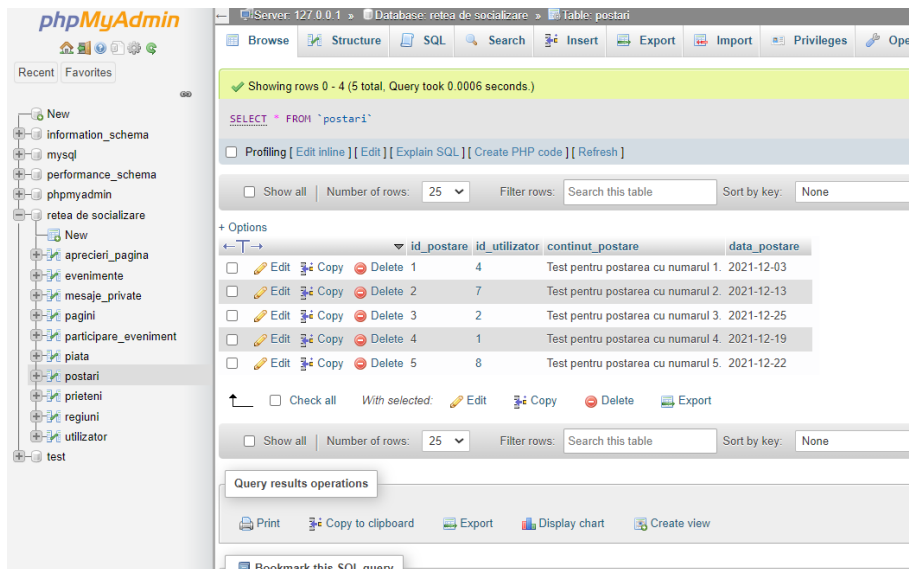
```
(1, 4, 'Test pentru postarea cu numarul 1.', '2021-12-03'),
```

```
(2, 7, 'Test pentru postarea cu numarul 2.', '2021-12-13'),
```

```
(3, 2, 'Test pentru postarea cu numarul 3.', '2021-12-25'),
```

```
(4, 1, 'Test pentru postarea cu numarul 4.', '2021-12-19'),
```

```
(5, 8, 'Test pentru postarea cu numarul 5.', '2021-12-22');
```



The screenshot shows the phpMyAdmin interface. On the left is the database navigation tree with 'postari' selected under 'retea de socializare'. The main panel displays the 'postari' table structure and data. The table has 5 rows of test data. Below the table are options for query results operations like Print, Copy to clipboard, Export, Display chart, and Create view.

	id_postare	id_utilizator	continut_postare	data_postare
<input type="checkbox"/>	1	4	Test pentru postarea cu numarul 1.	2021-12-03
<input type="checkbox"/>	2	7	Test pentru postarea cu numarul 2.	2021-12-13
<input type="checkbox"/>	3	2	Test pentru postarea cu numarul 3.	2021-12-25
<input type="checkbox"/>	4	1	Test pentru postarea cu numarul 4.	2021-12-19
<input type="checkbox"/>	5	8	Test pentru postarea cu numarul 5.	2021-12-22

Inserare MESAJE_PRIVATE

```
INSERT INTO `mesaje_private` (`id_mesaj`, `id_emitator`, `id_destinatar`, `continut_mesaj`, `data_mesaj`) VALUES
```

```
(1, 2, 3, 'Test pentru mesajul 1.', '2021-12-09'),
```

```
(2, 3, 2, 'Test pentru mesajul 2.', '2021-12-09'),
```

```
(3, 6, 1, 'Test pentru mesajul 3.', '2021-11-02'),
```

```
(4, 1, 6, 'Test pentru mesajul 4.', '2021-11-02'),
```

```
(5, 8, 9, 'Test pentru mesajul 5.', '2021-12-11');
```

Showing rows 0 - 4 (5 total. Query took 0.0007 seconds)

```
SELECT * FROM `mesaje_private`
```

	id_mesaj	id_emitator	id_destinatar	continut_mesaj	data_mesaj
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	2	3	Test pentru mesajul 1.	2021-12-09
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	3	2	Test pentru mesajul 2.	2021-12-09
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	6	1	Test pentru mesajul 3.	2021-11-02
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	1	6	Test pentru mesajul 4.	2021-11-02
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	8	9	Test pentru mesajul 5.	2021-12-11

Inserare PIATA

INSERT INTO `piata` (`id_produs`, `id_utilizator`, `nume_produs`, `descriere_produs`, `pret_produs`) VALUES

(1, 4, 'Produs 1', 'Descriere produs 1', 200),

(2, 1, 'Produs 2', 'Descriere produs 2', 5000),

(3, 6, 'Produs 3', 'Descriere produs 3', 10),

(4, 9, 'Produs 4', 'Descriere produs 4', 75),

(5, 3, 'Produs 5', 'Descriere produs 5', 90);

Showing rows 0 - 4 (5 total. Query took 0.0006 seconds)

```
SELECT * FROM `piata`
```

	id_produs	id_utilizator	nume_produs	descriere_produs	pret_produs
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	4	Produs 1	Descriere produs 1	200
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	1	Produs 2	Descriere produs 2	5000
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	6	Produs 3	Descriere produs 3	10
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	9	Produs 4	Descriere produs 4	75
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	3	Produs 5	Descriere produs 5	90

Inserare EVENIMENTE

```
INSERT INTO `evenimente` (`id_eveniment`, `titlu_eveniment`, `descriere_eveniment`,  
`data_incepere`, `data_incheiere`) VALUES
```

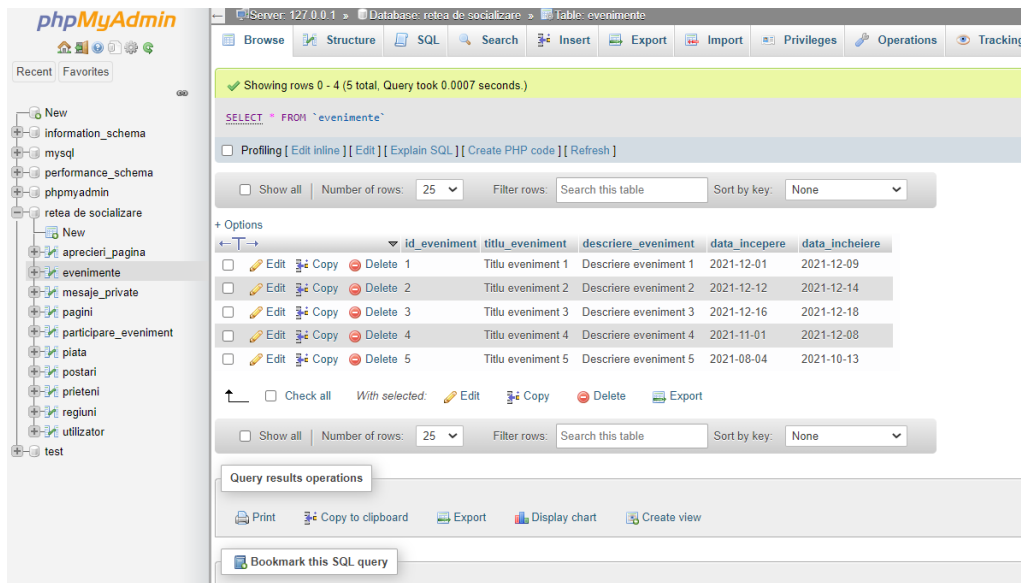
(1, 'Titlu eveniment 1', 'Descriere eveniment 1', '2021-12-01', '2021-12-09'),

(2, 'Titlu eveniment 2', 'Descriere eveniment 2', '2021-12-12', '2021-12-14'),

(3, 'Titlu eveniment 3', 'Descriere eveniment 3', '2021-12-16', '2021-12-18'),

(4, 'Titlu eveniment 4', 'Descriere eveniment 4', '2021-11-01', '2021-12-08'),

(5, 'Titlu eveniment 5', 'Descriere eveniment 5', '2021-08-04', '2021-10-13');



The screenshot shows the phpMyAdmin interface for a database named 'retea de socializare'. The 'evenimente' table is selected, and the 'Table: evenimente' tab is active. The table contains 5 rows of data. The columns are: id_eveniment, titlu_eveniment, descriere_eveniment, data_incepere, and data_incheiere. The data is as follows:

id_eveniment	titlu_eveniment	descriere_eveniment	data_incepere	data_incheiere
1	Titlu eveniment 1	Descriere eveniment 1	2021-12-01	2021-12-09
2	Titlu eveniment 2	Descriere eveniment 2	2021-12-12	2021-12-14
3	Titlu eveniment 3	Descriere eveniment 3	2021-12-16	2021-12-18
4	Titlu eveniment 4	Descriere eveniment 4	2021-11-01	2021-12-08
5	Titlu eveniment 5	Descriere eveniment 5	2021-08-04	2021-10-13

Inserare PARTICIPARE_EVENTIMENT

```
INSERT INTO `participare_eveniment` (`id_utilizator`, `id_eveniment`) VALUES
```

(1, 2),

(1, 4),

(1, 5),

(3, 1),

(3, 2),

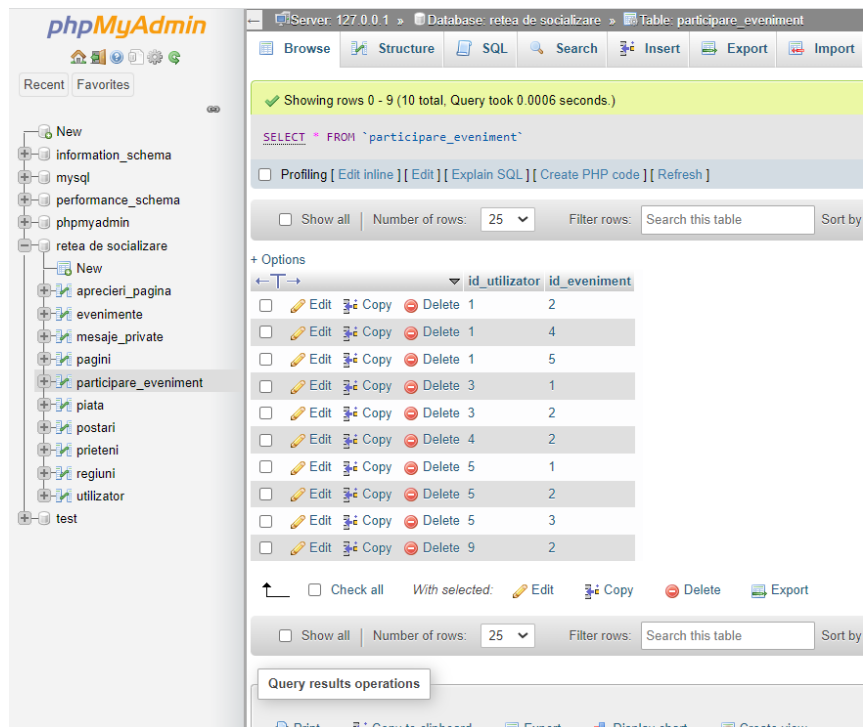
(4, 2),

(5, 1),

(5, 2),

(5, 3),

(9, 2);



Inserare PRIETENI

INSERT INTO `prieteni` (`id_utilizator1`, `id_utilizator2`, `raspuns`) VALUES

(1, 2, 1),

(1, 3, 1),

(1, 4, 1),

(1, 5, 1),

(1, 6, 0),

(1, 7, 0),

(1, 8, 1),

(1, 9, 0),

(1, 10, 1),

(2, 5, 1);

The screenshot shows the phpMyAdmin interface for the 'retea de socializare' database. The 'prieteni' table is selected, and its structure is displayed. The table has three columns: 'id_utilizator1', 'id_utilizator2', and 'raspuns'. The data is as follows:

	id_utilizator1	id_utilizator2	raspuns
<input type="checkbox"/>	1	2	1
<input type="checkbox"/>	1	3	1
<input type="checkbox"/>	1	4	1
<input type="checkbox"/>	1	5	1
<input type="checkbox"/>	1	6	0
<input type="checkbox"/>	1	7	0
<input type="checkbox"/>	1	8	1
<input type="checkbox"/>	1	9	0
<input type="checkbox"/>	1	10	1
<input type="checkbox"/>	2	5	1

Inserare APRECIERI_PAGINA

INSERT INTO `aprecieri_pagina` (`id_pagina`, `id_utilizator`) VALUES

(1, 1),

(1, 2),

(1, 3),

(1, 4),

(1, 5),

(2, 7),

(3, 9),

(4, 1),

(4, 2),

(5, 3);

The screenshot shows the phpMyAdmin interface for the 'retea de socializare' database. The 'aprecieri_pagina' table is selected, and its structure is displayed. The table has two columns: 'id_pagina' and 'id_utilizator'. The data is as follows:

	id_pagina	id_utilizator
<input type="checkbox"/>	1	1
<input type="checkbox"/>	1	2
<input type="checkbox"/>	1	3
<input type="checkbox"/>	1	4
<input type="checkbox"/>	1	5
<input type="checkbox"/>	2	7
<input type="checkbox"/>	3	9
<input type="checkbox"/>	4	1
<input type="checkbox"/>	4	2
<input type="checkbox"/>	5	3