

# **Evidența populației la poliție**

## **Proiect Baze de date**

Procop Iuliana

# CUPRINS

1. Descrierea cerințelor .....	3
2. Etapa de proiectare .....	4
3. Descrierea tabelor și a relațiilor dintre tabele .....	4
4. Constrângerile de integritate impuse .....	8
5. Funcționarea aplicației.....	10
6. Interogări simple .....	13
7. Interogări complexe .....	14

## 1. Descrierea cerințelor

Un student care a terminat Academia Poliției Române este repartizat în regiunea Dobrogei pentru a se ocupa de evidența populației. Odată numit în această poziție, acesta a constatat faptul că gestiunea populației este susținută folosind o bază de date, care conține toate informațiile necesare despre fiecare individ. Întreținerea datelor folosind o bază de date reprezintă una dintre cele mai bune modalități de stocare, care reduce semnificativ timpul și asigură corectitudinea datelor. Astfel, în urma observării funcționalității, acesta este pus să se ocupe de gestiunea activității poliției rutiere.

- Poliția rutieră se ocupă de evidența populației, care săvârșește diferite acțiuni ce sunt sancționate de poliție.
- Poliția deține informații legate de fiecare individ din regiunea Dobrogei. Pentru fiecare individ se rețin următoarele date: nume, prenume, CNP, sex, data nașterii.
- Fiecare persoană are un domiciliu care este reprezentat de: județ, localitate, stradă și număr.
- De asemenea, un individ poate fi proprietarul mai multor mașini, iar pentru fiecare mașină al fiecărui individ sunt prezente următoarele: număr de înmatriculare, firma mașinii și modelul acesteia.
- Poliția rutieră deține cazierul fiecărui individ, care este emis la o anumită oră și dată și are ca scop reținerea infracțiunilor. De asemenea, există diferite tipuri de infracțiuni, cum ar fi: penale și civile, datele acestea reprezentând denumirea.

## 2. Etapa de proiectare

Individ
Nume
Prenume
CNP
Sex
Data nașterii

Domiciliu
Județ
Localitate
Stradă
Număr

Mașină
Nr. înmatriculare
Firma
Model

Cazier
Data
Ora

Infracțiune
Denumire

## 3. Descrierea tabelelor și a relațiilor dintre tabele

Individ
Individ_ID <b>CP</b>
Nume
Prenume
CNP
Sex
Data nașterii

Domiciliu
Domiciliu_ID <b>CP</b>
Județ
Localitate
Stradă
Număr

Mașină
Mașină_ID <b>CP</b>
Nr. înmatriculare
Firma
Model

Cazier
Cazier_ID <b>CP</b>
Data
Ora

Infracțiune
Infracțiune_ID <b>CP</b>
Denumire

	<b>Individ</b>	<b>Domiciliu</b>	<b>Mașină</b>	<b>Cazier</b>	<b>Infracțiune</b>
<b>Individ</b>		<b>1:N</b>	<b>1:1</b>	<b>1:1</b>	
<b>Domiciliu</b>	<b>1:1</b>				
<b>Mașină</b>	<b>1:N</b>				
<b>Cazier</b>	<b>1:1</b>				<b>1:N</b>
<b>Infracțiune</b>				<b>1:N</b>	

Relațiile între tabelele bazei de date a poliției rutiere sunt:

**Individ – Domiciliu:**  $1:1 + 1:N = 1:N$

**Individ – Mașină:**  $1:N + 1:1 = 1:N$

**Individ – Cazier:**  $1:1 + 1:1 = 1:1$

**Cazier – Infracțiune:**  $1:N + 1:N = N:N$

## **Individ**

Individ – Individ: ----

Individ – Domiciliu: Un individ poate avea un singur domiciliu, deci din perspectiva tabelului Individ, există o relație de tip 1:1.

Individ – Mașină: Un individ poate fi proprietarul mai multor mașini, deci din perspectiva tabelului Individ, este o relație de tip 1:N.

Individ – Cazier: Un individ poate avea un singur cazier, iar din perspectiva tabelului Individ, reprezintă o relație 1:1.

## **Domiciliu**

Domiciliu – Individ: În cadrul unui singur domiciliu pot face parte mai mulți indivizi, astfel din perspectiva tabelului Domiciliu, reprezintă o relație de tip 1:N.

Domiciliu – Mașină: ---

Domiciliu – Cazier: ---

Domiciliu – Infracțiune: ---

## **Mașină**

Mașină – Indivd: O mașină poate avea un singur proprietar, deci din perspectiva tabelului Mașină, există o relație de tip 1:1.

Mașină – Domiciliu: ---

Mașină – Mașină: ---

Mașină – Cazier: ---

Mașină – Infracțiune: ---

## **Cazier**

Cazier – Indivd: Un cazier este asociat unui singur individ, un cazier deținând informații doar despre un singur individ. Astfel, relația dintre ele este de tip 1:1.

Cazier – Domiciliu: ---

Cazier – Mașină: ---

Cazier – Cazier: ---

Cazier – Infracțiune: Un cazier deține informații despre toate infracțiunile realizate de fiecare individ, deci din perspectiva tabelului Cazier, există o relație de tip 1:N.

## **Infracțiune**

Infracțiune – Indivd: ---

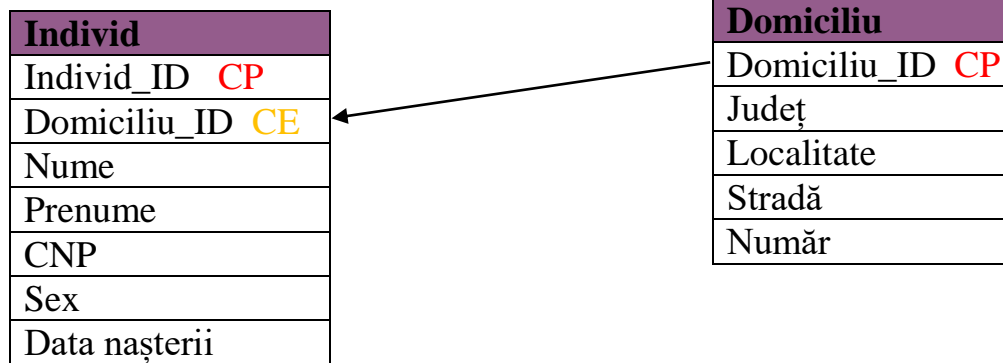
Infracțiune – Domiciliu: ---

Infracțiune – Mașină: ---

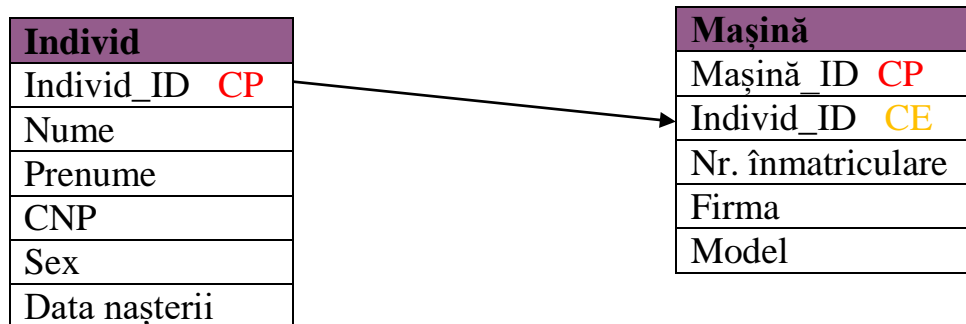
Infracțiune – Cazier: O infracțiune poate fi în mai multe cazieri, deci din perspectiva tabelului Tip infracțiune, există o relație de tip 1:N.

Infracțiune – Infracțiune: ---

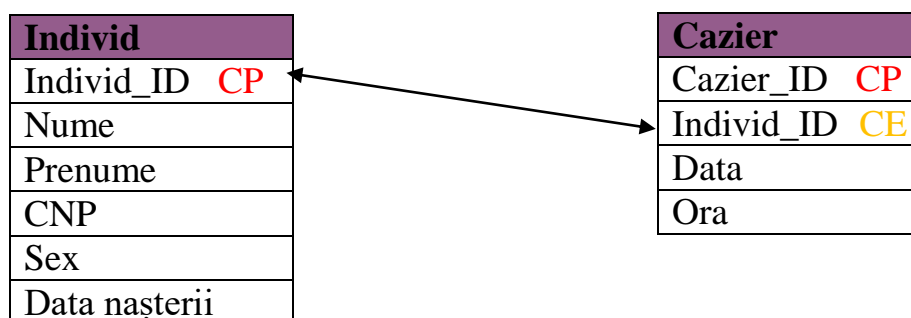
✓ **Individ – Domiciliu:** 1:1 + 1:N = **1:N**



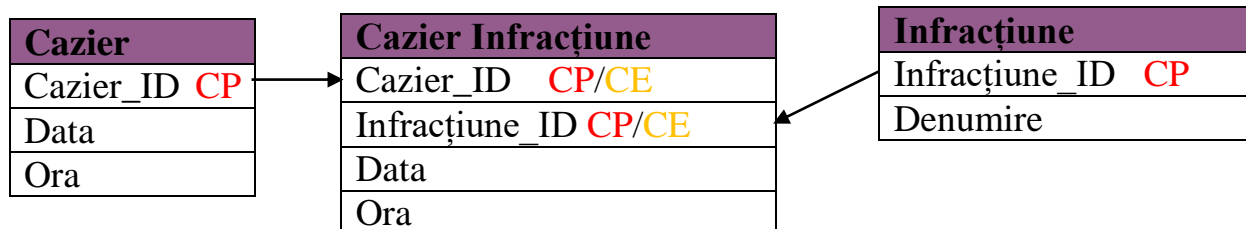
✓ **Individ – Mașină:** 1:N + 1:1 = **1:N**



✓ **Individ – Cazier:** 1:1 + 1:1 = **1:1**



✓ **Cazier – Infracțiune:** 1:N + 1:N = N:N



#### 4. Constrângerile de integritate impuse

1. Tabelul **Individ** are următoarele tabele: Individ\_ID, Domiciliu\_ID, Nume, Prenume, CNP, Sex, Data nașterii și constrângerile aferente:

PRIMARY KEY: Individ\_ID (cu autoincrementare)

FOREIGN KEY: Domiciliu\_ID

NOT NULL: Nume, Prenume, CNP

UNIQUE: CNP

CHECK: Sex ('M', 'F')

DEFAULT: Sex ('F')

2. Tabelul **Domiciliu** are următoarele tabele: Domiciliu\_ID, Județ, Localitate, Stradă, Număr și constrângerile aferente:

PRIMARY KEY: Domiciliu\_ID (cu autoincrementare)

NOT NULL: Județ, Localitate, Stradă, Număr



3. Tabelul **Mașină** are următoarele tabele: Mașină\_ID, Indivîd\_ID, Nr. înmatriculare, Firma, Model și constrângerile aferente:

PRIMARY KEY: Mașină\_ID (cu autoincrementare)

FOREIGN KEY: Indivîd\_ID

NOT NULL: Nr. înmatriculare , Firma, Model

UNIQUE: Nr. înmatriculare

4. Tabelul **Cazier** are următoarele tabele: Cazier\_ID, Indivîd\_ID, Data, Ora și constrângerile aferente:

PRIMARY KEY: Cazier\_ID (cu autoincrementare)

FOREIGN KEY: Indivîd\_ID

NOT NULL: Data, Ora

5. Tabelul **Infracțiune** are următoarele tabele: Infracțiune\_ID, Denumire și constrângerile aferente:

PRIMARY KEY: Infracțiune\_ID (cu autoincrementare)

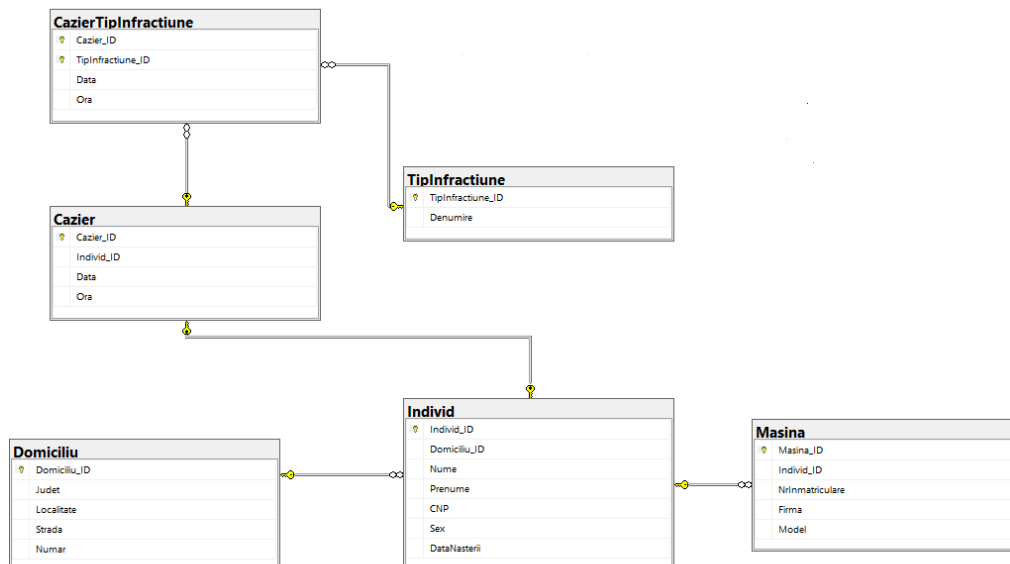
NOT NULL: Denumire

6. Tabelul **Cazier Infracțiune** are următoarele tabele: Cazier\_ID, Infracțiune\_ID, Data, Ora și constrângerile aferente:

PRIMARY KEY: Cazier\_ID + Infracțiune\_ID (cheie primară compusă)

FOREIGN KEY: Cazier\_ID + Infracțiune\_ID

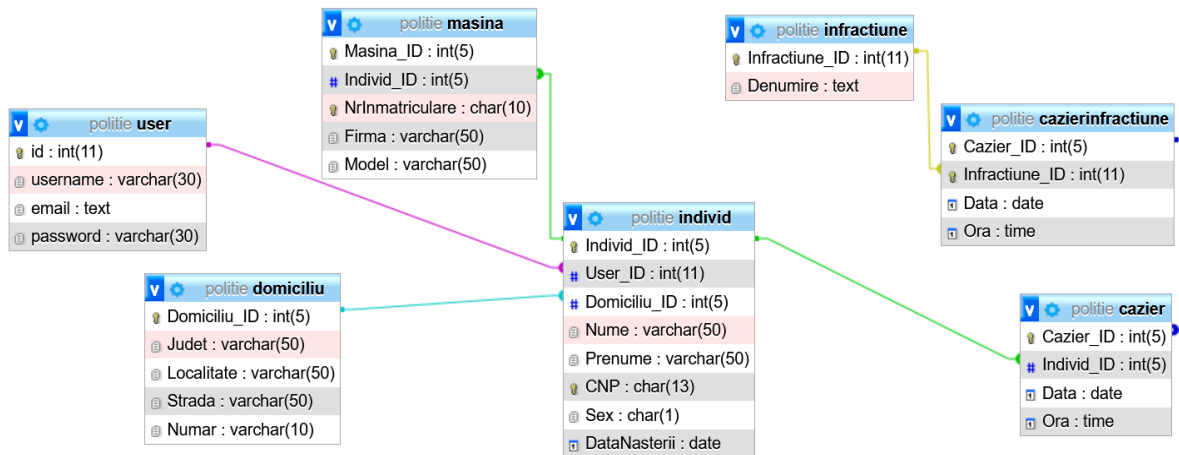
NOT NULL: Data, Ora



## 5. Funcționarea aplicației

În urma proiectării interfeței vizuale, implementarea acesteia într-un mediu de programare și corelarea cu baza de date, am introdus o tabelă pentru administratorul bazei de date, cu ajutorul căreia se realizează partea de login și de înregistrare (în cazul în care nu există un cont). Astfel, am adăugat în baza de date tabela aceasta, suferind modificările următoare:

User
User_ID CP
Username
Email
Password



Aplicația a fost realizată în phpmyadmin cu php și html.

Conectarea la baza de date se realizează în fișierul “conn.php”, astfel:

```

1  <?php
2
3  $sname= "localhost";
4  $uname= "root";
5  $password = "";
6
7  $db_name = "politie";
8
9  $conn = mysqli_connect($sname, $uname, $password, $db_name);
10
11  if (!$conn) {
12      echo "Connection failed!";
13  }
14  ?>
  
```

În cazul în care nu există un cont se realizează partea de înregistrare, unde sunt adăugate datele despre contul administratorului (“register.php”).

După conectarea la baza de date cu partea de login (“login.php”) se ajunge la pagina (“page.php”) de unde pot fi accesate datele despre fiecare tabelă din baza de date printr-un buton, de asemenea și rezultatele în urma interogărilor simple și complexe, dar și modificările aduse tabelelor individ și domiciliu.

Astfel, prin apăsarea primelor șase butoane se realizează accesarea datelor din fiecare tabelă a bazei de date, surprinsă în exemplul de mai jos (“individ.php”, “domiciliu.php”, “masina.php”, “cazier.php”, “infractiune.php”, “cazierinfractiune.php”).

```
indiv.php x domiciliu.php x masina.php x cazier.php x infractiune.php x cazierinfractiune.php x update.php x update_domiciliu.php x join.php x join2.php x sub
1 <?php
2 session_start();
3 include "conn.php";
4
5 $sql = "SELECT Individ_ID, User_ID, Domiciliu_ID, Nume, Prenume, CNP, Sex, DataNasterii FROM individ";
6 $result = $conn->query($sql);
7
8 if ($result->num_rows > 0) {
9
10     while($row = $result->fetch_assoc()) {
11         echo "ID: " . $row["Individ_ID"]. "<br>" . " User_ID: " . $row["User_ID"]. "<br>" . " Domiciliu_ID: " . $row["
            Domiciliu_ID"]. "<br>" . " Numele: " . $row["Nume"]. "<br>" . " Prenumele: " . $row["Prenume"]. "<br>" . " CNP:
            " . $row["CNP"]. "<br>" . " Sex: " . $row["Sex"]. "<br>" . " DataNasterii: " . $row["DataNasterii"]. "<br>" .
            "<br>";
12     }
13 } else {
14     echo "0 results";
15 }
16 $conn->close();
17 >>
18
19
20 <html>
21 <body>
22     <a href="page.php">Back</a>
23
24 </body>
25 </html>
26
```

Următoarele două butoane sunt pentru modificarea tabelii de indivizi și a tabelii ce conține datele despre domiciliul indivizilor. Modificările care se realizează pentru tabela de indivizi sunt: delete la datele individului după numele și prenumele acestuia și update la data nașterii a individului (în cazul în care a fost introdusă greșit) în funcție de numele și prenumele acestuia ("update.php"). De asemenea, pentru tabela domiciliu se realizează: insert pentru toate câmpurile tabelii domiciliu, delete la datele domiciliului în funcție de numele județului și numele localității și update la numărul străzii în funcție de numele localității și numele străzii ("update\_domiciliu.php").

Ultimele 3 butoane: "Cautare date personale indivizi" folosit pentru primele 3 interogări simple descrise mai jos, "Cautare date publice indivizi", folosit pentru următoarele 3 interogări simple și "Cautare particulara indivizi" pentru interogările complexe.

## 6. Interogări simple

JOIN-uri:

1. Nume și prenume indivizi care au mașină cu firma precizată de utilizator:

```
$sql = "SELECT individ.Nume, individ.Prenume, masina.Firma
        FROM individ RIGHT JOIN masina ON individ.Individ_ID = masina.Individ_ID
        WHERE masina.Firma = '$id'";
$result = $conn->query($sql);
```

2. Firma și modelul mașinii a individului cu numele precizat de utilizator:

```
$sql1 = "SELECT masina.Firma, masina.Model, individ.Nume
        FROM masina INNER JOIN individ ON masina.Individ_ID = individ.Individ_ID
        WHERE individ.Nume = '$id1'";
$result1 = $conn->query($sql1);
```

3. Nume și prenume indivizi care fac parte din județul precizat de utilizator:

```
$sql2 = "SELECT individ.Nume, individ.Prenume
        FROM individ INNER JOIN domiciliu ON
        individ.Domiciliu_ID = domiciliu.Domiciliu_ID
        WHERE domiciliu.Judet = '$id2'";
$result2 = $conn->query($sql2);
```

4. Nume și prenume indivizi care au mai mult de o mașină și au sexul dat de utilizator:

```
$sql3 = "SELECT individ.Nume, individ.Prenume
        FROM individ INNER JOIN masina ON
        individ.Individ_ID = masina.Individ_ID
        WHERE individ.Sex = '$id3'
        GROUP BY individ.Nume, individ.Prenume
        HAVING COUNT(masina.Masina_ID) > 1";

$result3 = $conn->query($sql3);
```

5. Nume și prenume indivizi care au cazierul emis după ora dată de utilizator:

```
$sql4 = "SELECT individ.Nume, individ.Prenume
        FROM individ INNER JOIN cazier ON
        individ.Individ_ID = cazier.Individ_ID
        WHERE cazier.Ora > '$id4'";

$result4 = $conn->query($sql4);
```

6. Nume și prenume indivizi care au cazierul emis înainte de data scrisă de utilizator:

```
$sql5 = "SELECT individ.Nume, individ.Prenume from individ INNER JOIN cazier ON
        individ.Individ_ID = cazier.Individ_ID
        WHERE cazier.Data < '$id5'";

$result5 = $conn->query($sql5);
```

## 7. Interogări complexe

### Subcereri:

1. Firma și modelul mașinii al căror proprietar(individ) are cazier emis după data introdusă:

```
$sql = "SELECT masina.Firma, masina.Model
        FROM masina
        WHERE masina.Individ_ID IN (SELECT individ.Individ_ID FROM individ, cazier
        WHERE individ.Individ_ID = cazier.Individ_ID
        AND cazier.Data > '$id')";

$result = $conn->query($sql);
```

2. Nume și prenume individ care nu are mașini cu firma mașinii dată de utilizator:

```
$sql1 = "SELECT individ.Nume, individ.Prenume
        FROM individ
        WHERE individ.Individ_ID NOT IN (SELECT masina.Individ_ID FROM individ, masina
                                         WHERE individ.Individ_ID = masina.Individ_ID
                                         AND masina.Firma = '$id1')";

$result1 = $conn->query($sql1);
```

3. Firma și modelul mașinii unde proprietarul are cazier emis înainte de ora specificată:

```
$sql2 = "SELECT masina.Firma, masina.Model
        FROM masina
        WHERE EXISTS (SELECT individ.Individ_ID FROM individ, cazier
                      WHERE individ.Individ_ID = cazier.Individ_ID
                      AND cazier.Ora < '$id2')";

$result2 = $conn->query($sql2);
```

4. Nume și prenume indivizi care sunt mai tineri decât toți indivizii din localitatea:

```
$sql3 = "SELECT individ.Nume, individ.Prenume
        FROM individ
        WHERE individ.DataNasterii > ALL(SELECT individ.DataNasterii FROM individ, domiciliu
                                         WHERE individ.Domiciliu_ID = domiciliu.Domiciliu_ID
                                         AND domiciliu.Localitate = '$id3')";

$result3 = $conn->query($sql3);
```