***UNIVERSITATEA “ȘTEFAN CEL MARE”, SUCEAVA***

***FACULTATEA DE INGINERIE ELECTRICĂ ȘI ȘTIINȚA CALCULATOARELOR***

***SPECIALIZAREA CALCULATOARE***

***AN 2, GRUPA 3124 A***

**PROIECT  DISCIPLINA PIU**

**Aplicație Gestiune Farmacie Lavinia**

**(C#)**

**Autor: Cozma Lavinia-Cristina**

**Profesor coordonator: Ovidiu Gherman**

2025

**TEMĂ  PROIECT**

**Descriere temă**

Aplicație de gestiune a Farmaciei Lavinia are următoarele opțiuni disponibile:

Angajat:

* adăugare medicament;
* modificare medicament;
* actualizare medicamente;
* cautare medicament;
* ștergere medicament.

Client:

* căutare medicament;
* finalizare cumpărături;

***De ce am ales tema?***

Am ales să realizez proiectul pe această temă deoarece domeniul farmaceutic este un domeniu foarte des întâlnit, este actual și util, iar aplicația mi-a oferit posibilitatea de a aplica într-un mod practic și ușor toate conceptele învățate, dar și să înțeleg modul de gestionare al unei farmacii.

**CUPRINS**

**CAPITOLUL I**

Descriere proiect din punct de vedere al utilizatorului…………..….………………………….4

**CAPITOLUL II**

Descriere proiect din punct de vedere al programatorului……..….………………………10

**CAPITOLUL III**

Dezvoltari ulterioare………………………………….……..….…………..…………………………………….….12

Bibliografie......................................................................................................................12

**CAPITOLUL I**

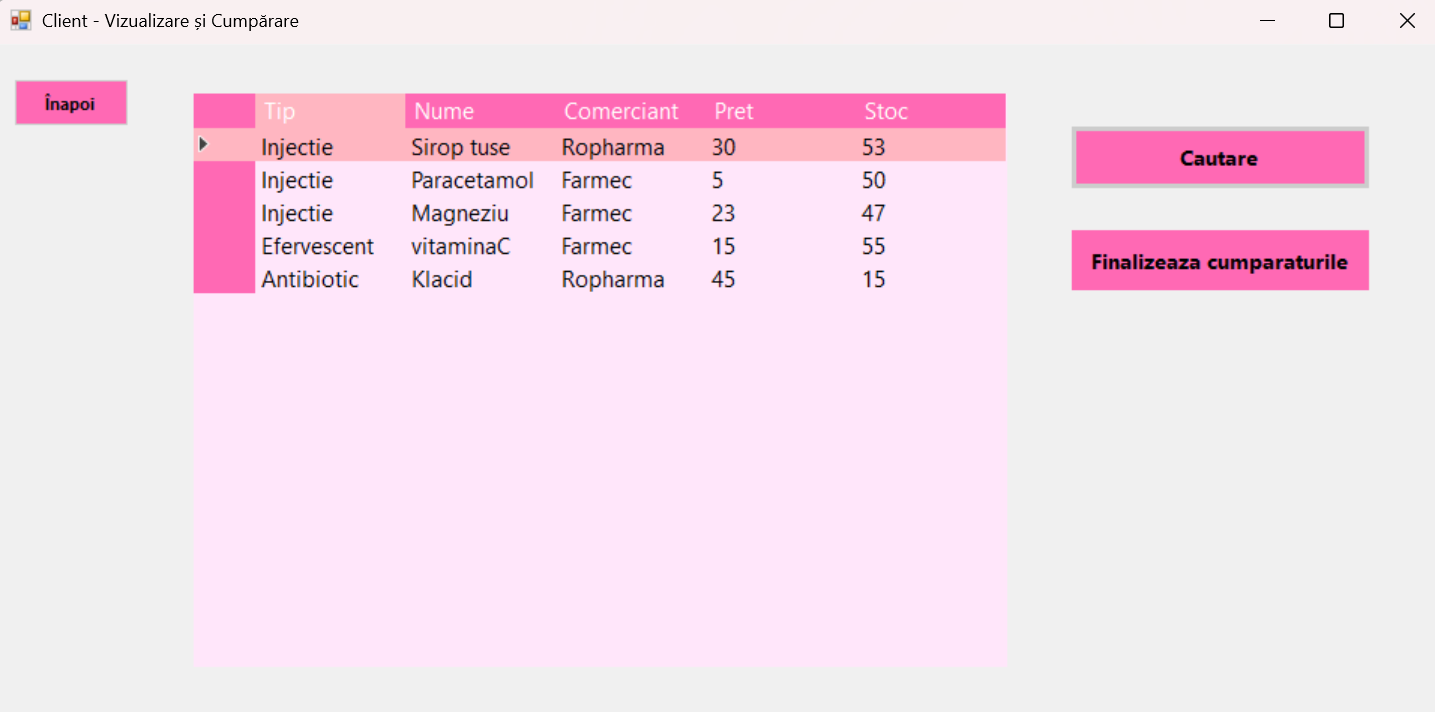
**Descriere proiect din punct de vedere al utilizatorului**

Interfața cu uitilizatorul este prezentată mai jos:

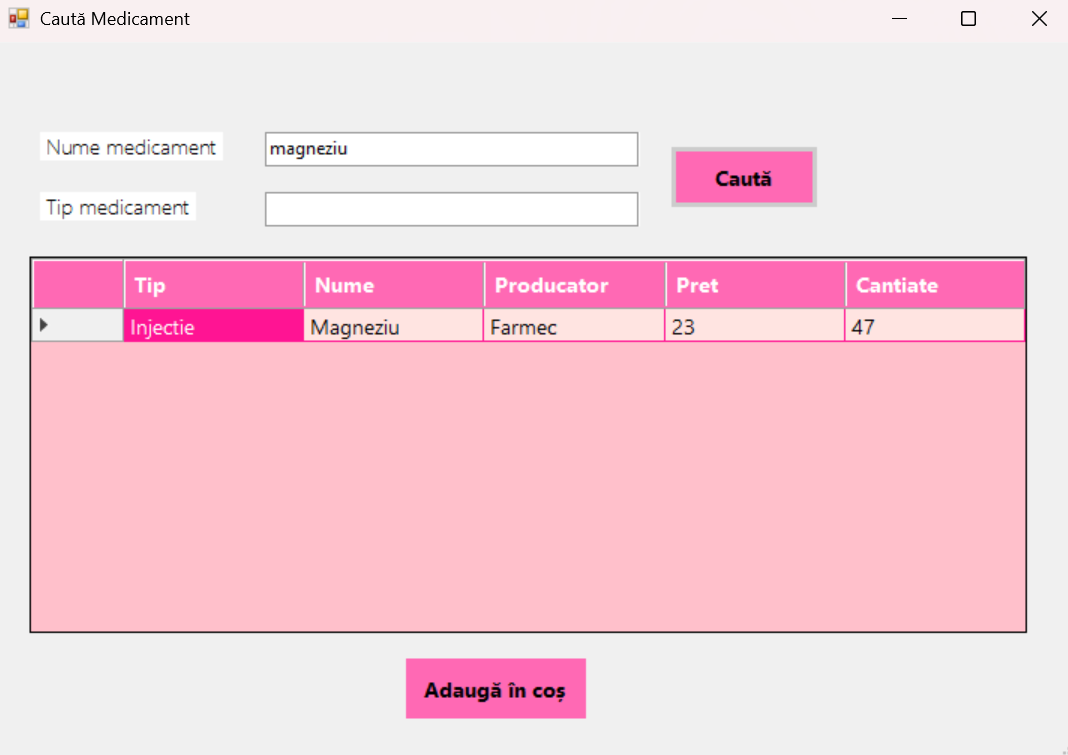
Meniul principal are 2 opțiuni: Angajat și Client.

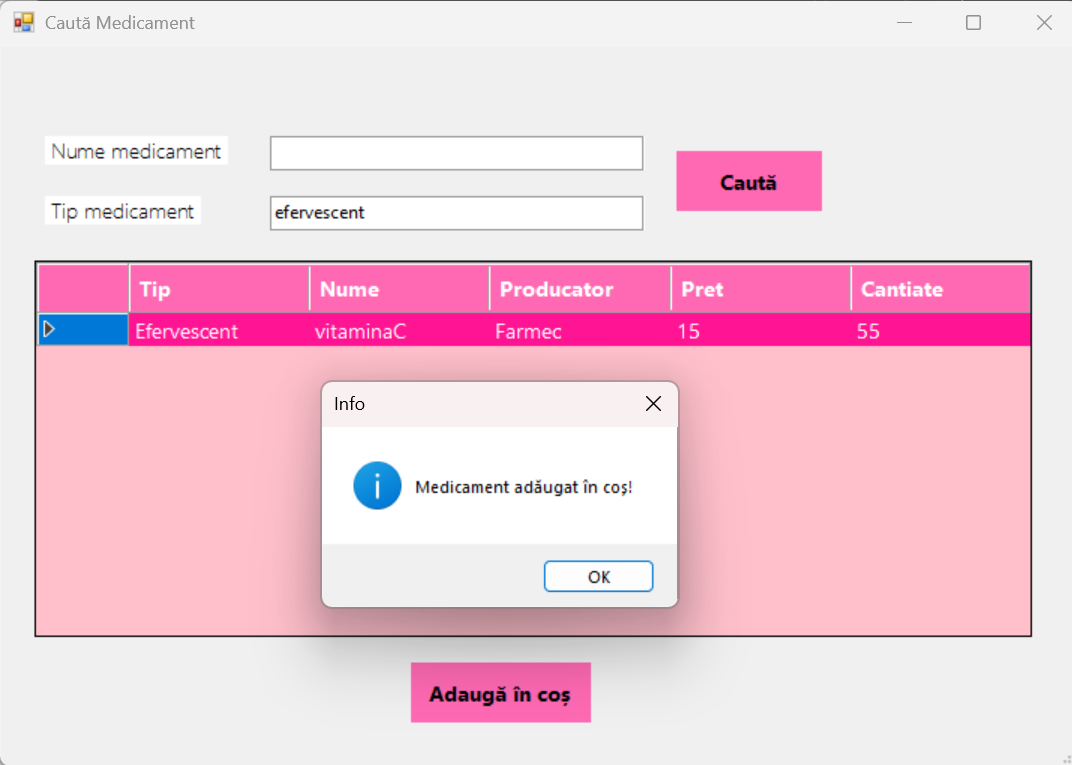


Formularul pentru client conține 2 opțiuni: căutare și finalizează cumpărăturile.

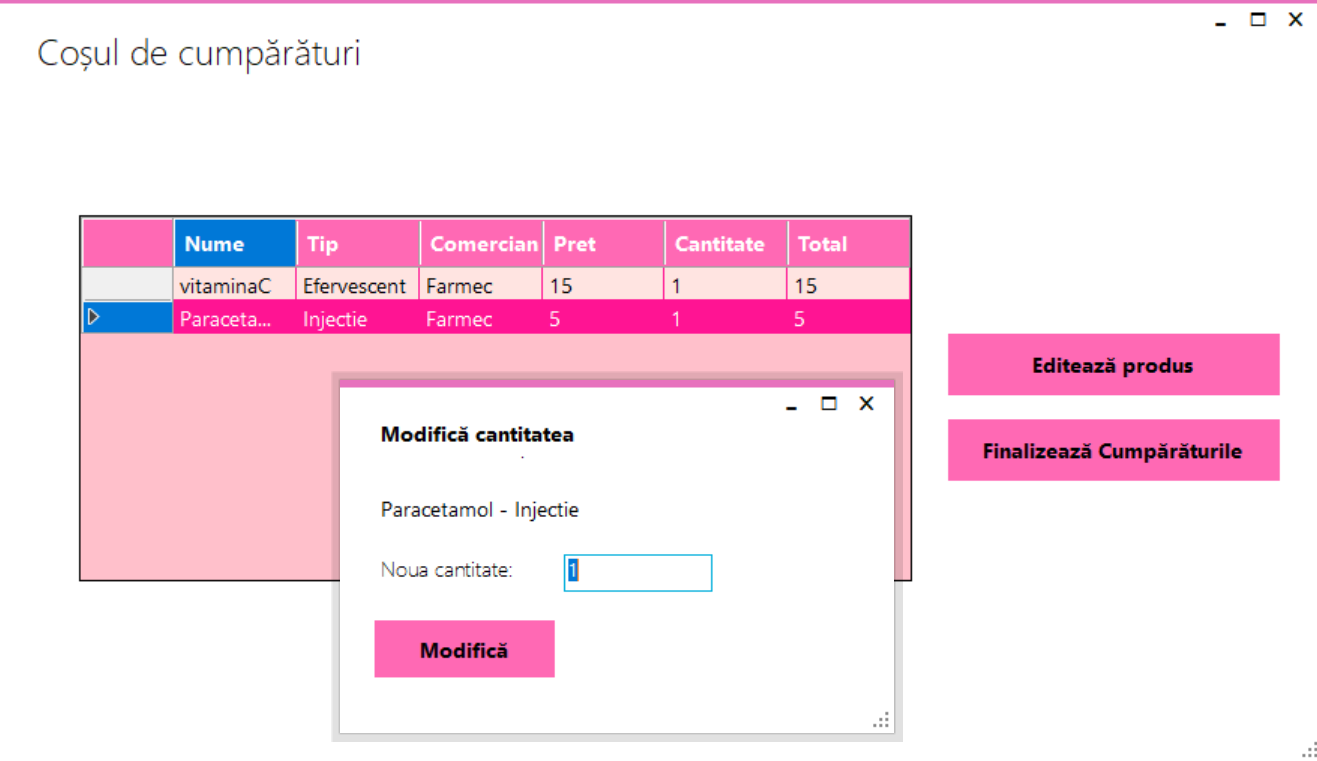


Pentru a căuta un medicament, clientul dispune de două modalități de căutare: după numele medicamentului și după tipul medicamentului. După ce clientul a găsit medicamentul căutat, acesta are posibilitatea de a-l adăuga în coș.

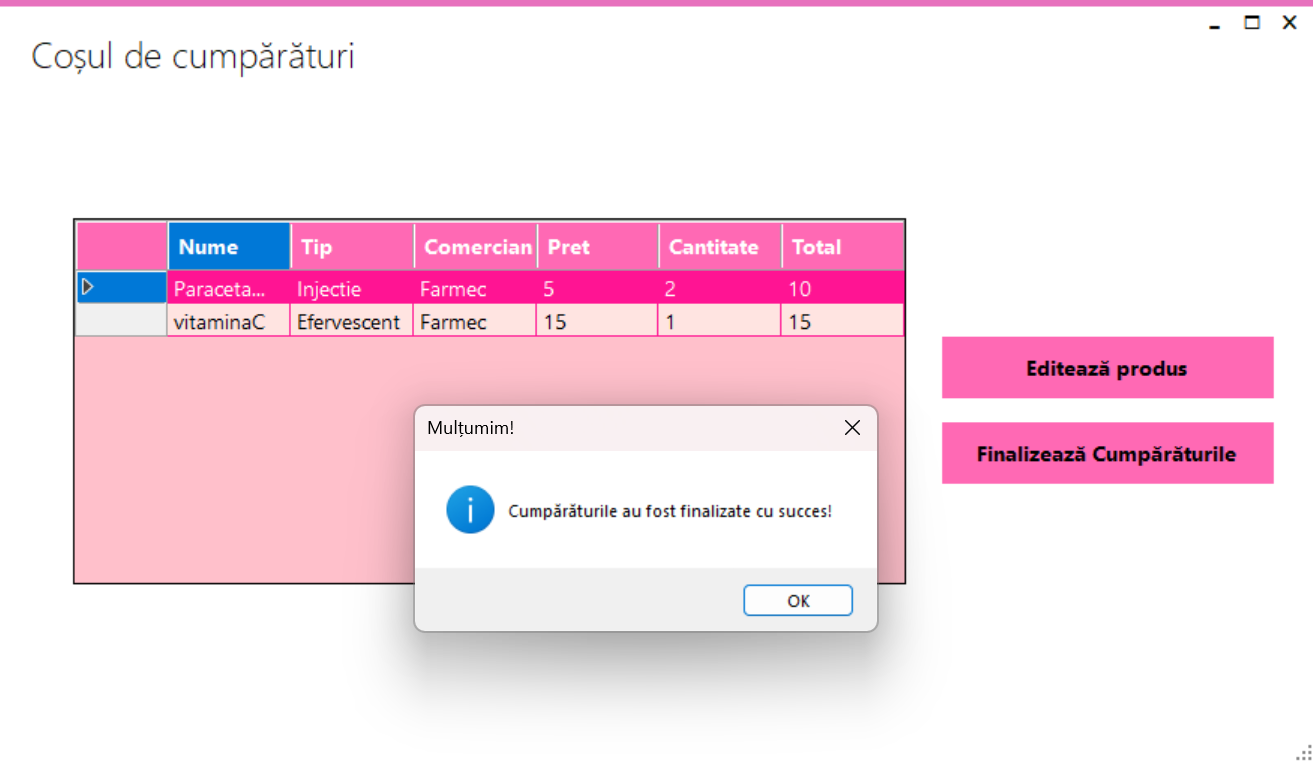




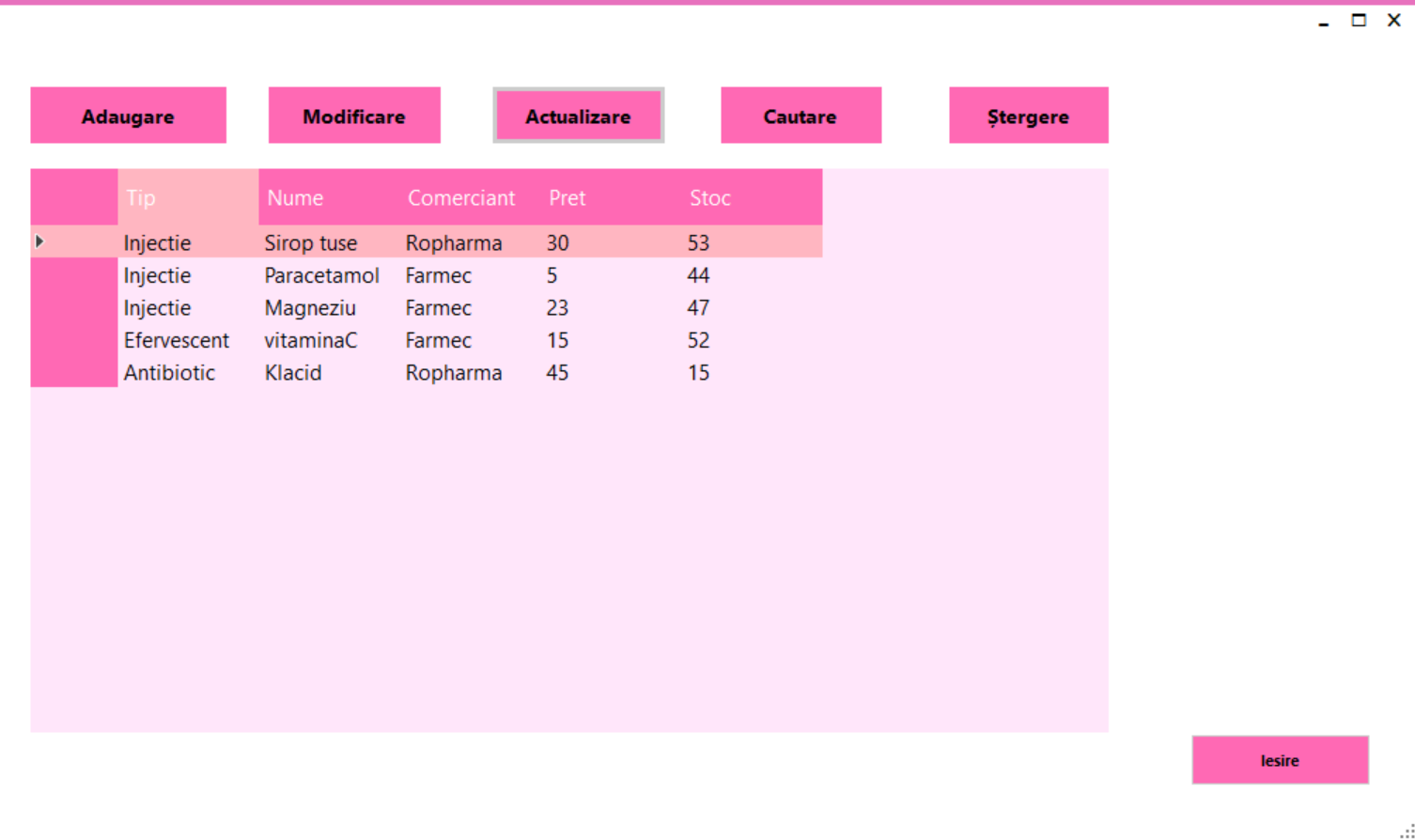
Înainte de a finaliza cumpărăturile, clientul are posibilitatea de a edita produsul, adică de a adăuga mai multe cantități pentru un medicament dorit.



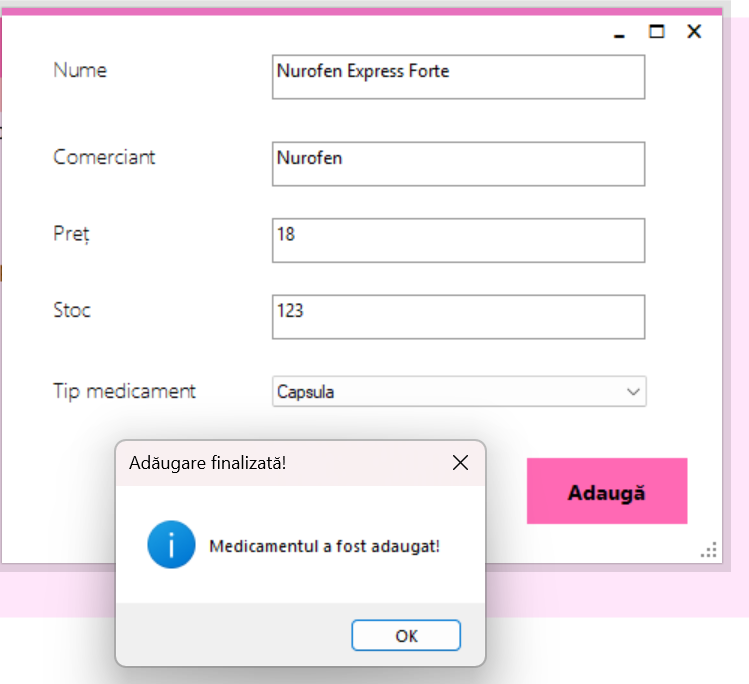
La finalizarea cumpărăturilor, clientul primește un mesaj pentru confirmare.



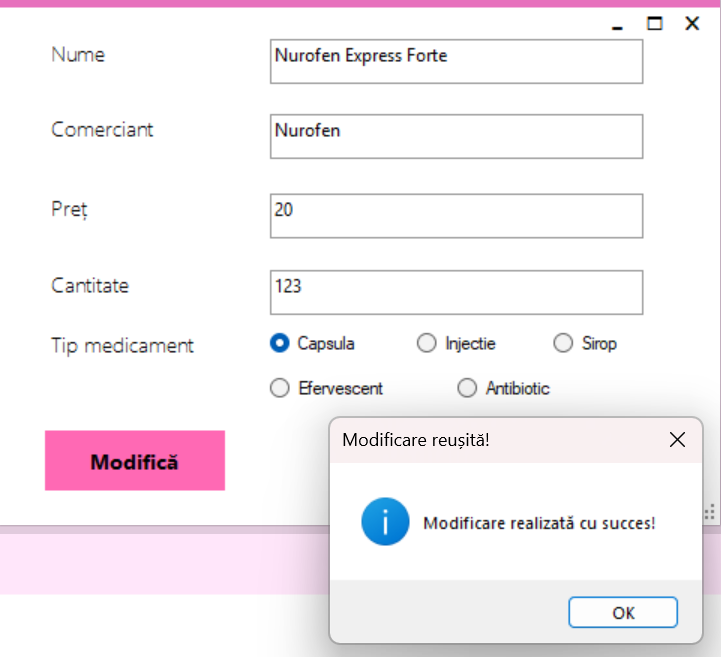
Formularul pentru angajat îi oferă acestuia posibilitatea de a se conecta farmacie cu beneficii avansate față de un client obișnuit. Angajatul poate adăuga un medicament, modifica un medicament,actualiza lista de medicamente, căuta un medicament, și de a șterge medicamente.



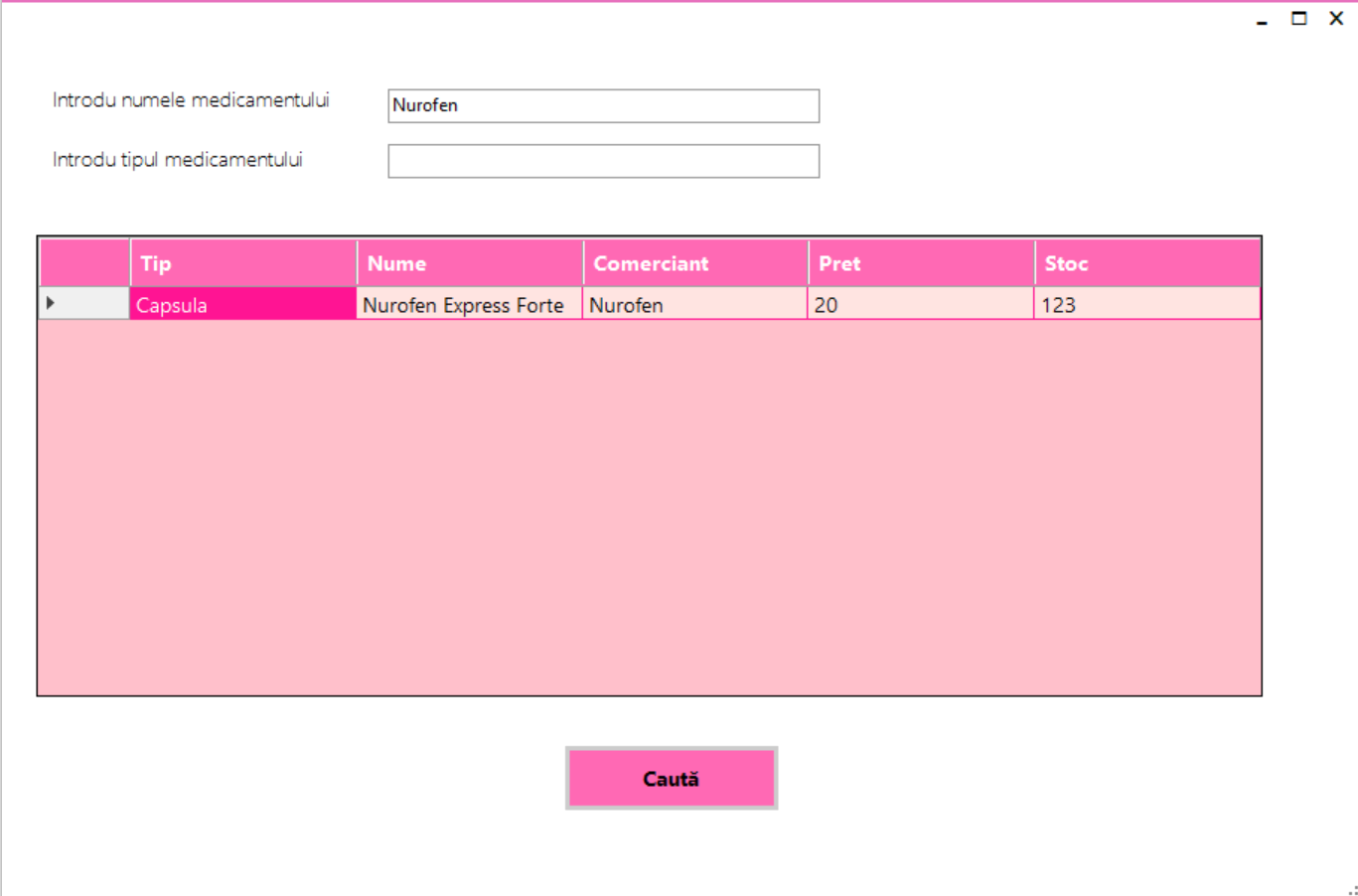
Pentru a adăuga un medicament, angajatul trebuie sa completeze câteva informații valide pentru a putea finaliza adăugarea.

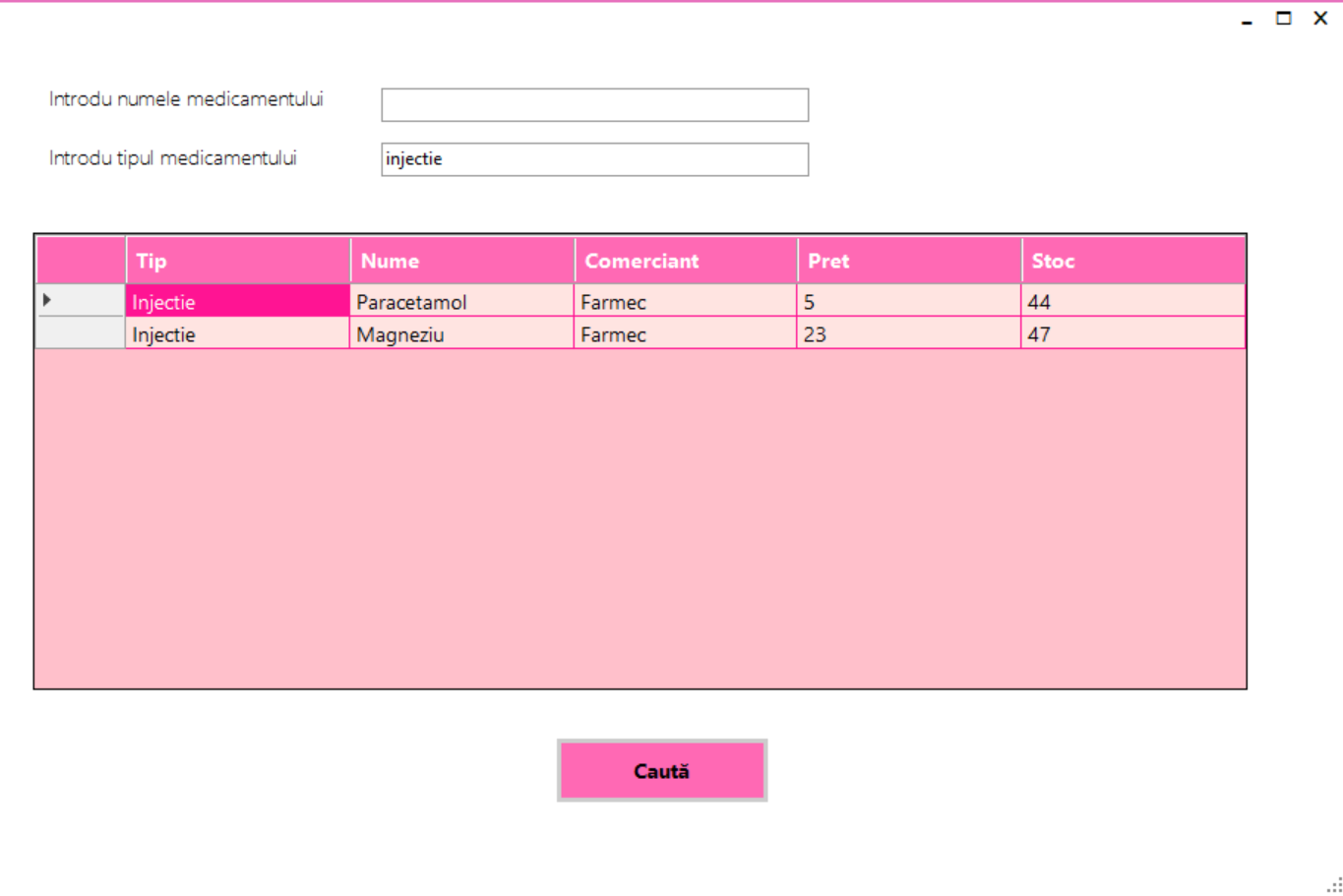


Modificarea medicamentului se poate face pentru orice informație pentru un medicament ales de angajat.



Căutarea unui medicament se poate face după două modalități de căutare: după numele medicamentului și după tipul medicamentului.



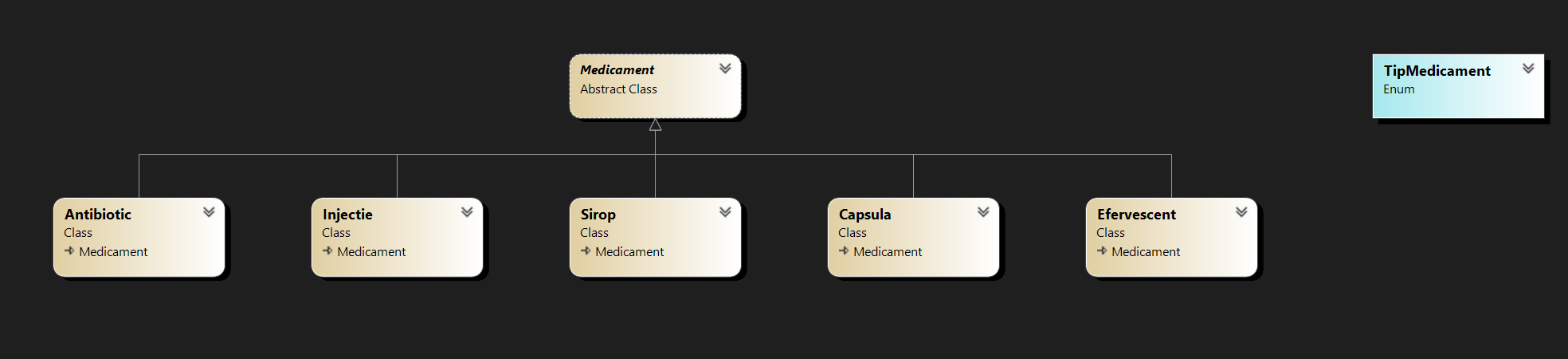


Angajatorul mai are posibilitatea de a șterge un medicament selectat și de a actualiza lista pentru medicamentele existente în acel moment în farmacie.

**CAPITOLUL II**

**Descriere proiect din punct de vedere al programatorului**

* Diagrama de clase



* Descrierea claselor

Clasa *TipMedicament* este o enumerare și reprezintă tipurile de medicamente disponibile: capsulă, injecție, sirop, efervescent și antibiotic.

Clasa *Medicament* este o clasă abstractă și reprezintă o clasa de bază pentru clasele *Antibiotic, Injectie, Sirop, Capsula, Efervescent* și definește proprietățile comune pentru toate medicamentele: nume, comerciant, preț, stoc, tip și metodele: ToString() care afișează medicamentul într-un mod mai ușor de citit și ToFileFormat() care returneaza o linie pentru salvarea în fișier a medicamentului.

Clasa *Antibiotic* mostenește clasa Medicament și transmite datele către clasa de bază.

Clasa *Injectie* mostenește clasa Medicament și transmite datele către clasa de bază.

Clasa *Sirop* mostenește clasa Medicament și transmite datele către clasa de bază.

Clasa *Capsula* mostenește clasa Medicament și transmite datele către clasa de bază.

Clasa *Efervescent* mostenește clasa Medicament și transmite datele către clasa de bază.

* Sectiuni de cod deosebite

În aplicația mea am decis sa folosesc biblioteca MetroFramework pentru a crea un UI cât mai prietenos și intuitiv, cu funcții clare și o experiență cât mai placută și mai eficientă.

**CAPITOLUL III**

**Dezvoltări ulterioare**

* Functionalități noi care ar fi necesare:

- Aplicația ar putea avea un mod de înregistrare și logare atât pentru angajat, cât și pentru client;

- Clientul ar putea avea acees, pe viitor, la un istoric al comenzilor și posibilitatea primirii notificărilor în cazul în care un medicament a apărut din nou pe stoc.

* Modificări de cod sau de tehnologii de implementare

- În viitor aș dori sa includ în proiect o bază de date și să dezvolt mai mult design-ul farmaciei mele.

**BIBLIOGRAFIE**

* Documentație Laborator
* Documentație Curs
* Chat GPT 4o