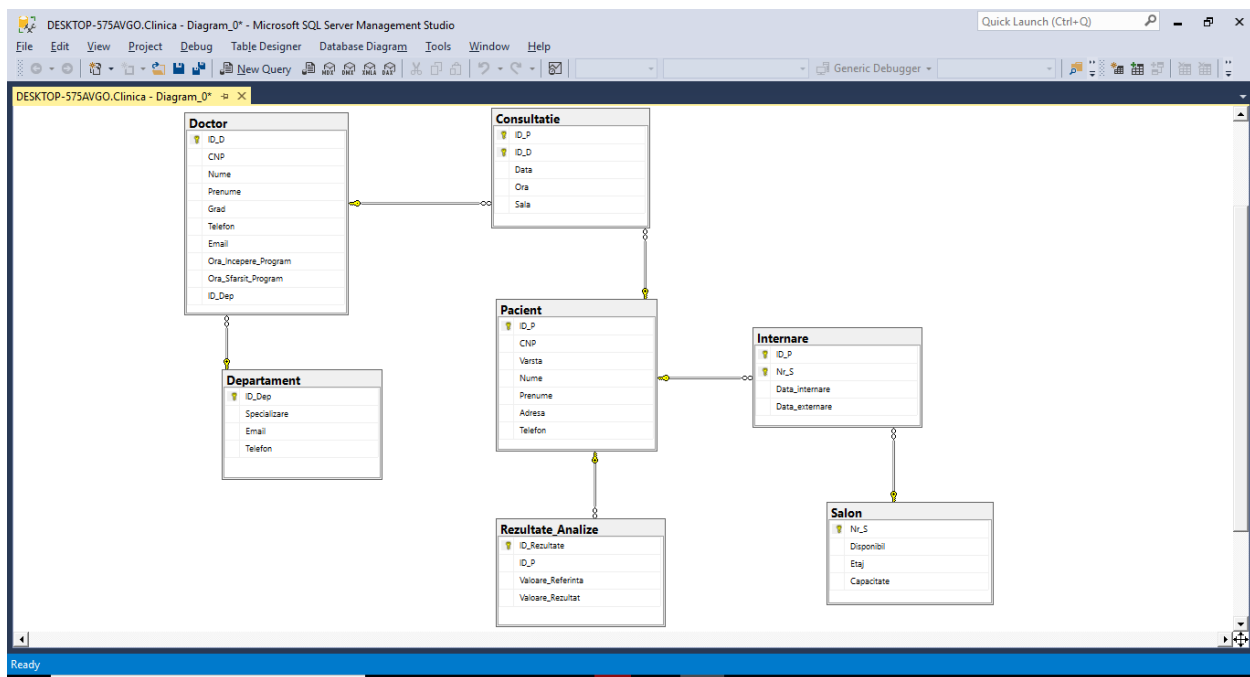


Tema proiectului meu este: *“Gestiunea consultatiilor la o clinica medicala”*

### Descrierea cerintelor:

Pentru gestiunea consultatiilor in cadrul unei clinici medicale, avem nevoie sa stocam pacientii consultati sau internati, doctorii, saloanele din cadrul clinicii, rezultatele analizelor in cadrul clinicii, departamentele medicale cu doctori in cadrul clinicii, internarile din clinica si consultatiile din clinica. Dorim ca baza de date sa fie utila atat personalului din cadrul clinicii, pentru a gestiona consultatiile si internarile din cadrul clinicii, cat si pacientilor care doresc a afla anumite informatii despre: doctorii din cadrul clinicii, saloanele unde au fost internati, salile unde vor fi sau au fost consultati.

Pentru implementarea temei, in etapa de proiectare am realizat urmatoarea baza de date:



### Etapa de proiectare, descrierea tabelor si a relatiilor dintre ele, constrangerile de integritate impuse:

Baza de date pe care am gandit-o consta in 7 tabele (Pacient, Doctor, Consultatie, Salon, Internare, Rezultate\_Analize si Departament). Atributele alese pentru entitatile pe care le-am identificat sunt alese astfel incat sa se stocheze doar informatiile relevante.

Baza de date consta in doua relatii N:N pe care le-am transformat in relatii 1:N cu ajutorul unor entitati de intersectie si doua relatii 1:N. In final, rezulta 6 relatii 1:N, asa cum se poate observa si in schema de mai sus: Pacient:Rezultate\_Analize, Departament:Doctor, Doctor:Programare, Pacient:Programare, Pacient:Internare, Salon:Internare, unde fiecare entitate care reprezinta N-ul intr-o relatie va prelua cheia primara a entitatii care reprezinta 1, ca foreign key. Pentru 4 tabele (Pacient, Doctor, Departament, Rezultate\_Analize) am folosit chei primare artificiale, compuse dintr-un singur element, ID, iar pentru tabelele Consultatie si Internare am folosit chei primare compuse din cheile

primare artificiale preluate ca foreign key si attribute care sa diferentieze in mod unic inregistrarile intre ele(de exemplu, pentru Consultatie, unde cheia primara va fi ID\_P,ID\_D + Data, compunerea foreign key urilor ID\_P si ID\_D cu Data va da caracter unic cheii primare, deoarece doar un pacient poate fi consultat doar de un singur doctor la o singura data).

Tabelele Consultatie si Internare reprezinta tabele de intersectie pentru relatiile N:N Pacient:Doctor(un pacient poate fi consultat de mai multi doctori, iar un doctor poate consulta mai multi pacienti), respectiv Pacient:Salon(un pacient poate fi internat in mai multe saloane, iar un salon poate avea mai multi pacienti internati). Prin intermediul entitatii intersectie Consultatie, relatia N:N Pacient:Doctor se va imparti in doua relatii 1:N, Pacient:Consultatie, respectiv Doctor:Consultatie(asadar, entitatea Consultatie va prelua ID\_P si ID\_D ca foreign key, iar compunerea acestora cu Data internarii va reprezenta cheia primara unica a acestei entitati) si rezulta ca:

- o consultatie are un singur pacient, un pacient poate avea mai multe consultatii;
- o consultatie are un singur doctor, un doctor poate avea mai multe consultatii;

Prin intermediul entitatii intersectie Internare, relatia N:N Pacient:Salon se va imparti in doua relatii 1:N, Pacient:Internare, respective Pacient:Salon(asadar, entitatea Internare va prelua ID\_P si ID\_S ca foreign key, iar compunerea acestora cu Data\_internare si Data\_externare va reprezenta cheia primara unica a acestei entitati) si rezulta ca:

- o internare este intr-un singur salon, un salon poate avea mai multe internari;
- o internare are un singur pacient, un pacient poate avea mai multe internari;

Relatia Pacient:Rezultate\_Analize este 1:N(un pacient poate avea mai multe seturi de rezultate, dar un set de rezultate apartine unui singur pacient), unde entitatea Rezultate\_Analize preia ID\_P ca foreign key attributele entitatii Rezultate\_Analize, Valoare\_Referinta si Valoare\_Rezultat, reprezinta: valoarea de referinta pentru analizele de hemoglobina din sange, fata de care comparam valoarea rezultata pentru analizele unui pacient(cu cat este mai aproape de valoarea de referinta, cu cat rezultatele pacientului sunt mai bune).

Relatia Departament:Doctor este 1:N(un departament poate avea mai multi doctori, in timp ce un doctor face parte dintr-un singur departament medical.

#### Functionalitatea aplicatiei:

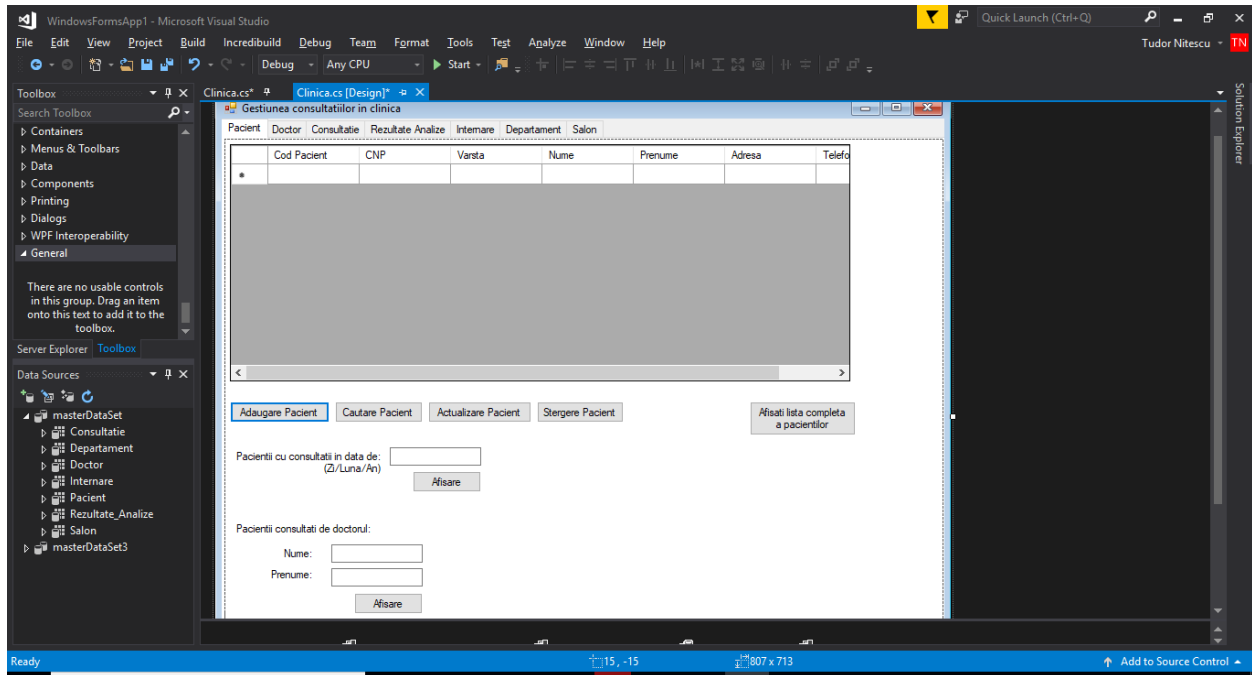
Baza de date am implementat-o in SQL Server 2017, iar aplicatia pentru interfata grafica am implementat-o in Visual Studio 2017(ca Windows Forms Application)

Aplicatia implementeaza entitatile descrise mai sus, cu attributele din baza de date din SQL Server(prin conectarea Visual Studio cu MS SQL Server), functii de adaugare, stergere, actualizare, afisare si cautare pentru pacient si doctor, functia de adaugare pentru consultatii, adaugare pentru rezultatele analizelor, adaugare de internari, adaugare, cautare, stergere, afisare si actualizare pentru departamentele medicale din cadrul clinicii, adaugare si stergere pentru saloane(in cazul in care intra in renovare/este scos din functiune/iese din renovare/este pus la loc in functionare). Pe langa aceste functii de baza, aplicatia implementeaza 8 query-uri simple si 4 query-uri complexe, pentru afisarea unor informatii relevante din cadrul clinicii medicale.

Aplicatia urmareste implementarea unor functii/cautari cat mai logice si mai relevante pentru clinica medicala(de exemplu, nu am implementat functia de stergere pentru consultatie sau internare, deoarece dorim sa tinem evidenta lor in permanenta).

Aplicatia are cate o fereastra(tab) pentru fiecare entitate, iar fiecare tab va avea cate un datagridview pentru afisarea rezultatelor query-urilor, sau afisarea listei complete de inregistrari.

De exemplu, pentru pacient:



In partea de sus putem selecta tab-ul corespunzator entitatii pe care dorim sa o accesam, in prima parte a paginii se poate observa datagridview-ul pentru pacient, iar sub acesta cateva butoane si textbox-uri pentru implementarea functiilor si query-urilor.

Toate paginile au o structura asemanatoare, pentru a facilita familiarizarea cu aplicatia.

Interogările simple:

1. Sa se afiseze doctorii care sunt din departamentul selectat de utilizator din droplist;

```
select Doctor.CNP, Doctor.ID_D, Doctor.Nume, Doctor.Prenume, Doctor.Grad, Doctor.Telefon,
Doctor.Email from Doctor INNER JOIN Departament ON Doctor.ID_Dep=Departament.ID_Dep where
Departament.Specializare="" + comboBox1.Text + ""
```

2. Sa se afiseze pacientii consultati in data introdusa de utilizator in textbox;

```
select Pacient.CNP, Pacient.ID_P, Pacient.Varsta, Pacient.Nume, Pacient.Prenume, Pacient.Adresa,
Pacient.Telefon from Pacient INNER JOIN Consultatie ON Pacient.ID_P=Consultatie.ID_P where
Consultatie.Data="" + textBox1.Text + ""
```

3. Sa se afiseze consultatiile doctorului cu numele si prenumele introduse de utilizator in textbox;

```
select Consultatie.Data, Consultatie.Ora, Consultatie.Sala from Consultatie INNER JOIN Doctor ON
Consultatie.ID_D=Doctor.ID_D where Doctor.Nume="" + textBox32.Text + "" and Doctor.Prenume="" +
textBox33.Text + ""
```

4. Sa se afiseze internarile din salonul selectat din droplist de catre utilizator;

```
select Internare.ID_P, Internare.Nr_S, Internare.Data_internare, Internare.Data_externare from
Internare INNER JOIN Salon ON Internare.Nr_S=Salon.Nr_S where Salon.Nr_S="" + comboBox2.Text + ""
```

5. Sa se afiseze internarile de la etajul introdus de utilizator in textbox;

```
select Internare.ID_P, Internare.Nr_S, Internare.Data_internare, Internare.Data_externare from
Internare INNER JOIN Salon ON Internare.Nr_S=Salon.Nr_S where Salon.Etaj="" + textBox2.Text + ""
```

6. Sa se afiseze saloanele cu internari in data introdusa de utilizator in textbox;

```
select Salon.Nr_S, Salon.Etaj from Salon INNER JOIN Internare ON Salon.Nr_S=Internare.Nr_S where
Internare.Data_Internare="" + textBox31.Text + ""
```

7. Sa se afiseze saloanele cu externari in data introdusa de utilizator in textbox;

```
select Salon.Nr_S, Salon.Etaj from Salon INNER JOIN Internare ON Salon.Nr_S=Internare.Nr_S where
Internare.Data_Externare="" + textBox34.Text + ""
```

8. Sa se afiseze rezultatele analizelor pentru pacientul cu CNP-ul introdus de utilizator in textbox;

```
select Rezultate_Analize.Valoare_Referinta, Rezultate_Analize.Valoare_Rezultat from Rezultate_Analize
INNER JOIN Pacient ON Rezultate_Analize.ID_P=Pacient.ID_P where Pacient.CNP="" + textBox28.Text + ""
```

Interogările complexe:

1. Sa se afiseze consultatiile din sala introdusa in textbox(dat de utilizator), ale doctorilor din departamentul selectat din combobox(selectat de catre utilizator din lista departamentelor medicale ale clinicii).

```
select Data, Ora, Sala from Consultatie where Consultatie.Sala="" + textBox6.Text + "" AND Consultatie.ID_D IN (select Doctor.ID_D from Doctor INNER JOIN Departament ON Doctor.ID_Dep = Departament.ID_Dep where Departament.Specializare = "" + comboBox4.Text + "")
```

2. Sa se afiseze pacientii doctorului cu numele si prenumele introduce de utilizator.

```
select distinct Pacient.CNP, Pacient.Nume, Pacient.Prenume, Pacient.Telefon from Pacient INNER JOIN Consultatie on Pacient.ID_P IN (select Consultatie.ID_P from Consultatie INNER JOIN Doctor on Consultatie.ID_D=Doctor.ID_D where Doctor.Nume="" + textBox29.Text + "" and Doctor.Prenume="" + textBox30.Text + "")
```

3. Sa se afiseze rezultatele analizelor pacientilor care au fost in sala introdusa de utilizator in textbox.

```
select distinct Rezultate_Analize.Valoare_Referinta, Rezultate_Analize.Valoare_Rezultat from Rezultate_Analize INNER JOIN Pacient on Rezultate_Analize.ID_P IN (select Pacient.ID_P from Pacient INNER JOIN Internare on Pacient.ID_P=Internare.ID_P where Nr_S="" + textBox35.Text + "")
```

4. Sa se afiseze pacientii consultati de doctorii din departamentul cu specializarea selectata de utilizator din combobox(droplist)

```
select distinct Pacient.CNP, Pacient.Nume, Pacient.Prenume, Pacient.Telefon from Pacient INNER JOIN Consultatie on Pacient.ID_P IN (select Consultatie.ID_P from Consultatie INNER JOIN Doctor on Consultatie.ID_D IN (select Doctor.ID_D from Doctor INNER JOIN Departament ON Doctor.ID_Dep=Departament.ID_Dep where Departament.Specializare = "" + comboBox3.Text + ""))
```