

Universitatea  
Transilvania  
din Braşov

FACULTATEA DE INGINERIE ELECTRICĂ  
ŞI ŞTIINŢA CALCULATOARELOR

# ***Dental Office***

## ***Membrii echipei:***

***Project manager:*** Ghiuzan Roxana-Otilia

***Developer:*** Călin Cristina-Ioana

***Tester:*** Puia Andrea

***Documentație:*** Bădiță Alexandru-Cristian

# **CUPRINS**

<b>Obiectivele proiectului</b>	<b>2</b>
<b>1.1 Introducere</b>	<b>2</b>
<b>1.2 Informații utile</b>	<b>2</b>
<b>1.3 Scopul proiectului</b>	<b>3</b>
<b>1.4 Obiective</b>	<b>3</b>
<b>Analiza de sistem</b>	<b>4</b>
<b>2.1 Metodologia</b>	<b>4</b>
<b>2.2 Aplicații similare</b>	<b>4</b>
<b>2.3 Sistemul propus</b>	<b>5</b>
<b>Design</b>	<b>5</b>
<b>3.1 Database Design</b>	<b>5</b>
<b>3.2 User Interface Design</b>	<b>10</b>
<b>Implementare</b>	<b>17</b>
<b>4.1 Mod de funcționare</b>	<b>17</b>
<b>4.2 Mod de realizare</b>	<b>17</b>
<b>Testare</b>	<b>18</b>
<b>5.1 Unit testing</b>	<b>18</b>
<b>5.2 Exemple Unit Test Case</b>	<b>19</b>

## **Capitol 1**

### **Obiectivele proiectului**

#### **1.1 Introducere**

Organizația Mondială a Sănătății definește stomatologia ca fiind “știința de a preveni, diagnostica și trata bolile, leziunile și malformațiile dinților, maxilarului și gurii”.

Stomatologii diagnostichează și tratează problemele dinților și țesuturilor din interiorul gurii. De asemenea, dau sfaturi de îngrijire a cavității bucale, pentru a ajuta la prevenirea viitoarelor probleme.

Ei oferă diferite servicii, precum punerea de plombe, pentru tratarea cariilor, extracții, albirea dinților, îndreptarea dinților, prin intermediul aparatelor dentare. De asemenea, se ocupă de implanturi dentare, pentru înlocuirea dinților lipsă.

Centrele de asistență medicală dentară sunt importante pentru că oferă servicii esențiale societății, așa că este necesar să își automatizeze procesul de lucru pentru a spori calitatea serviciilor oferite.

Din această cauză, este necesar să folosească un sistem de baze de date care să asigure stocarea datelor și să eficientizeze munca centrelor de asistență medicală dentară.

#### **1.2 Informații utile**

Utilizatorii ideali, ai aplicației, sunt cabinetele de asistență dentară, mai exact stomatologii. Aceștia, prin intermediul aplicației, își pot gestiona programările, serviciile oferite, precum și prețul acestora.

Interfața este intuitivă, întrucât este ușor de înțeles și de folosit. La început prezintă un ecran de logare, care solicită o adresă de utilizator și o parolă.

În continuare, după ce s-au introdus datele corect, se deschide meniul principal, ce îi permite medicului să introducă sau să editeze informații despre programări și servicii

De asemenea, există și un meniu de administrator, ce permite gestionarea ușoară a medicilor cabinetului. Accesul la acesta se realizează automat, atunci când aplicația detectează datele de conectare, ale unei persoane cu drepturi de administrator.

### **1.3 Scopul proiectului**

Scopul proiectului este crearea unui sistem de management care să îndeplinească toate nevoile precum înregistrarea doctorilor și pacienților, stabilirea programărilor și a serviciilor dentare oferite.

Fiind un sistem de management al unei clinici dentare, aplicația cuprinde patru mari module:

- I. Managementul medicilor
  - A. Adăugare medici
  - B. Ștergere medici
- II. Managementul serviciilor oferite
  - A. Adăugare servicii
  - B. Ștergere servicii
  - C. Modificarea serviciilor (costul acestora)
  - D. Verificarea istoricului de prețuri a serviciilor
- III. Managementul pacienților
  - A. Adăugare pacienți
  - B. Ștergerea pacienți
  - C. Stabilirea programărilor
  - D. Ștergerea programărilor
- IV. Rapoartele
  - A. Încasările aduse de fiecare medic
  - B. Gradul de ocupare a medicilor

### **1.4 Obiective**

1. Aplicația urmărește gestionarea programărilor pacienților la diferiți doctori.
2. Lista pacienților se actualizează prin adăugarea acestora de către doctori.
3. Intervențiile aferente fiecărui pacient includ costurile și data.
4. Rolul de administrator presupune crearea profilului de doctor și includerea intervențiilor disponibile la cabinet.

## **Capitol 2**

### **Analiza de sistem**

Analiza sistemului este un proces logic. Scopul acestei faze este de a determina ce trebuie făcut pentru a rezolva cerința. Obiectivul de bază al etapei de analiză este dezvoltarea modelului logic al sistemului.

#### **2.1 Metodologia**

Ciclul de viață al dezvoltării sistemelor este un model conceptual folosit în managementul proiectelor pentru a descrie etapele acestuia de dezvoltare a unui sistem până la finalizarea lui.

Metodologia pentru dezvoltarea acestui sistem este metoda cascadă. Vom folosi un mini interviu pentru a aduna datele necesare dezvoltării proiectului.

Întrebări:

1. Care este tipul de sistem utilizat?
2. Ce probleme avem întâlnite la alte sisteme?
3. Ce tipuri de pacienți avem?
4. Care sunt informațiile pe care le păstrăm?
5. Există un istoric?

#### **2.2 Aplicații similare**

##### **Solutionreach Dental**

Solutionreach Dental este un software de management al cabinetelor stomatologice cu accent pe interacțiunile dintre stomatolog și pacient. În acest sens platforma oferă ambelor părți un canal de comunicare în ceea ce privește programările și alte detalii stomatologice.

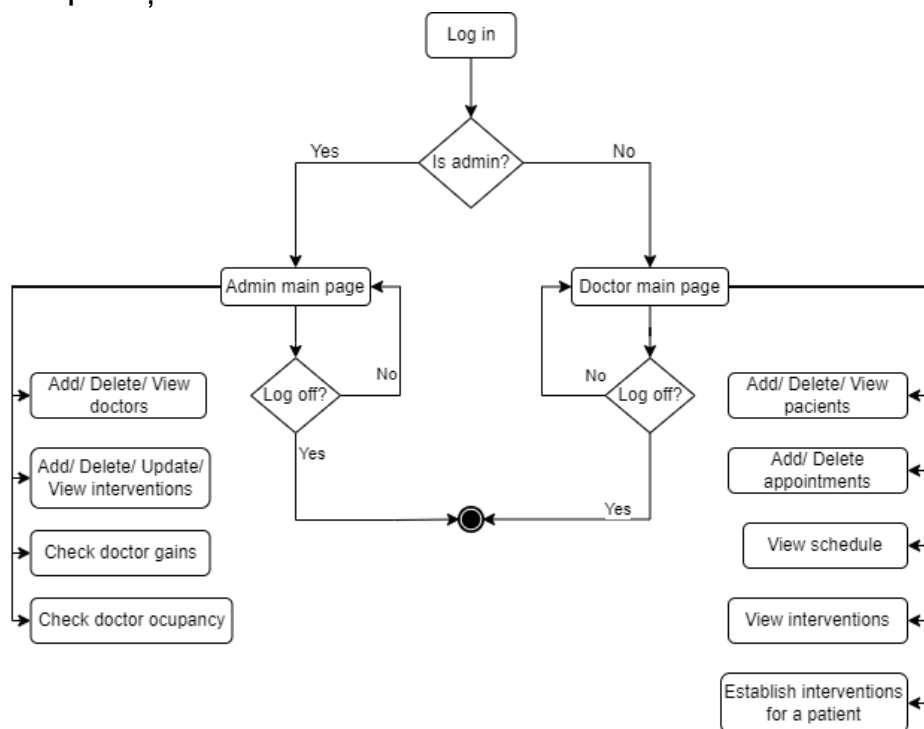
##### **Swell**

Swell este o platforma care permite efectuarea și trimiterea cererilor de programare. Pacienții pot indica ziua și ora la care doresc o programare la care medicul poate aproba în consecință.

Un punct de vedere din care această aplicație strălucește cu adevărat este rubrica de review-uri pentru monitorizarea reputației online a cabinetului.

## 2.3 Sistemul propus

Diagrama aplicației:



## Capitol 3

### Design

#### 3.1 Database Design

Baza de date este o colecție de informații care este organizată astfel încât să fie ușor accesată, gestionată și actualizată.

Diagramele E-R (Entitate-Asociere) sunt o reprezentare mai detaliată și logică a entităților, asocierilor și elementelor de date pentru o organizație.

Cele trei elemente principale ale diagramei sunt:

- Entitatea - persoană, loc, obiect, eveniment sau concept despre care datele urmează să fie păstrate
- Relația (asocierea) - este asocierea între instanțe ale unuia sau mai multor tipuri de entități și poate fi reprezentată ca opțional, obligatoriu mulți, obligatoriu unul și multe cu maximum (n)
- Atributul - numit proprietate sau caracteristică a unei entități

**Tabelele din baza de date:****1. User**

SQLQuery7.sql - CRI...ALIN\Cristina (68) X SQLQuery6.sql - CRI...ALIN\C

```

/***** Script for SelectTopNRows command from SSMS
SELECT TOP (1000) [ID]
      , [Username]
      , [Password]
      , [FirstName]
      , [LastName]
      , [Role]
      , [Deleted]
FROM [dentalOfficeDB].[dbo].[User]

```

100 %

Results Messages

	ID	Username	Password	FirstName	LastName	Role	Deleted
1	1	cris	q1w2e3	Cristina	Calin	administrator	no
2	2	andrea	q1w2e3	Andrea	Puia	administrator	no
3	3	baditaaex	q1w2e3	Alexandru	Badita	doctor	no
4	4	ghiuzanrx	q1w2e3	Roxana	Ghiuzan	doctor	yes
5	5	botschovi	q1w2e3	Ovidiu	Botsch	doctor	yes

**2. Pacient**

SQLQuery6.sql - CRI...ALIN\Cristina (66) X SQLQuery5.sql - CRI...ALIN\C

```

/***** Script for SelectTopNRows command from SSMS
SELECT TOP (1000) [ID]
      , [IDdoctor]
      , [FirstName]
      , [LastName]
      , [CNP]
      , [City]
      , [Deleted]
FROM [dentalOfficeDB].[dbo].[Patient]

```

100 %

Results Messages

	ID	IDdoctor	FirstName	LastName	CNP	City	Deleted
1	1	3	George	Dinu	2991017081811	Timisoara	yes
2	2	3	Andrea	Puia	2990606081811	Brasov	yes
3	3	3	Puia	Andrea	2991017081854	Brasov	yes
4	4	3	Vlad	Dinu	5000130801811	Ploiesti	no
5	5	3	Ovidiu	Botsch	1991017081811	Bucuresti	yes

### 3. Programări

SQLQuery5.sql - CRI...ALIN\Cristina (62)) X SQLQuery4.sql - CRI...ALIN\

```
/****** Script for SelectTopNRows command from SSMS
SELECT TOP (1000) [ID]
      , [IDpatient]
      , [Date]
      , [Hour]
      , [Deleted]
FROM [dentalOfficeDB].[dbo].[Appointment]
```

100 %

Results Messages

	ID	IDpatient	Date	Hour	Deleted
1	1	2	2021-12-16	9	yes
2	2	2	2021-12-17	14	yes
3	3	2	2021-12-16	11	yes
4	4	1	2021-12-15	12	yes
5	5	4	2022-01-05	10	yes
6	6	4	2022-01-06	12	no
7	7	4	2022-01-05	10	no

### 4. Intervenții

SQLQuery4.sql - CRI...ALIN\Cristina (53)) X

```
/****** Script for SelectTopNRows command from SSMS
SELECT TOP (1000) [ID]
      , [Name]
      , [Deleted]
FROM [dentalOfficeDB].[dbo].[Intervention]
```

100 %

Results Messages

	ID	Name	Deleted
1	1	plomba	no
2	2	consultatie	yes
3	3	anestezie	no
4	4	extractie	no
5	5	obturbatie	no
6	6	albire	no
7	7	al	yes
8	8	albb	yes
9	9	plomba2	yes
10	10	fcfcb	yes



## 5. Preț intervenție

SQLQuery8.sql - CRI...ALIN\Cristina (70) X SQLQuery7.sql - C

```

/***** Script for SelectTopNRows command fr
SELECT TOP (1000) [ID]
      , [IDintervention]
      , [Value]
      , [StartDate]
      , [EndDate]
      , [Deleted]
FROM [dentalOfficeDB].[dbo].[Price]
  
```

100 %

Results Messages

	ID	IDintervention	Value	StartDate	EndDate	Deleted
1	1	1	50.00	2021-11-23	2021-11-23	no
2	2	2	20.00	2021-11-25	2021-11-25	yes
3	3	3	30.50	2021-05-31	2021-12-30	no
4	4	4	60.70	2021-05-31	2021-12-30	no
5	5	5	190.00	2021-05-31	2021-12-30	no
6	6	6	50.00	2021-12-15	2021-12-15	no
7	7	4	56.00	2021-12-16	NULL	no
8	8	6	80.00	2021-12-15	2022-01-04	no
9	9	7	23.00	2021-12-16	2021-12-16	yes
10	10	8	55.00	2021-12-03	2021-12-03	yes
11	11	9	20.00	2022-01-07	2022-01-07	yes
12	12	6	34.00	2022-01-05	NULL	no
13	13	10	99.00	2022-01-06	2022-01-06	yes

## 6. Agendă

SQLQuery9.sql - CRI...ALIN\Cristina (71) X SQLQuery8.sql

```

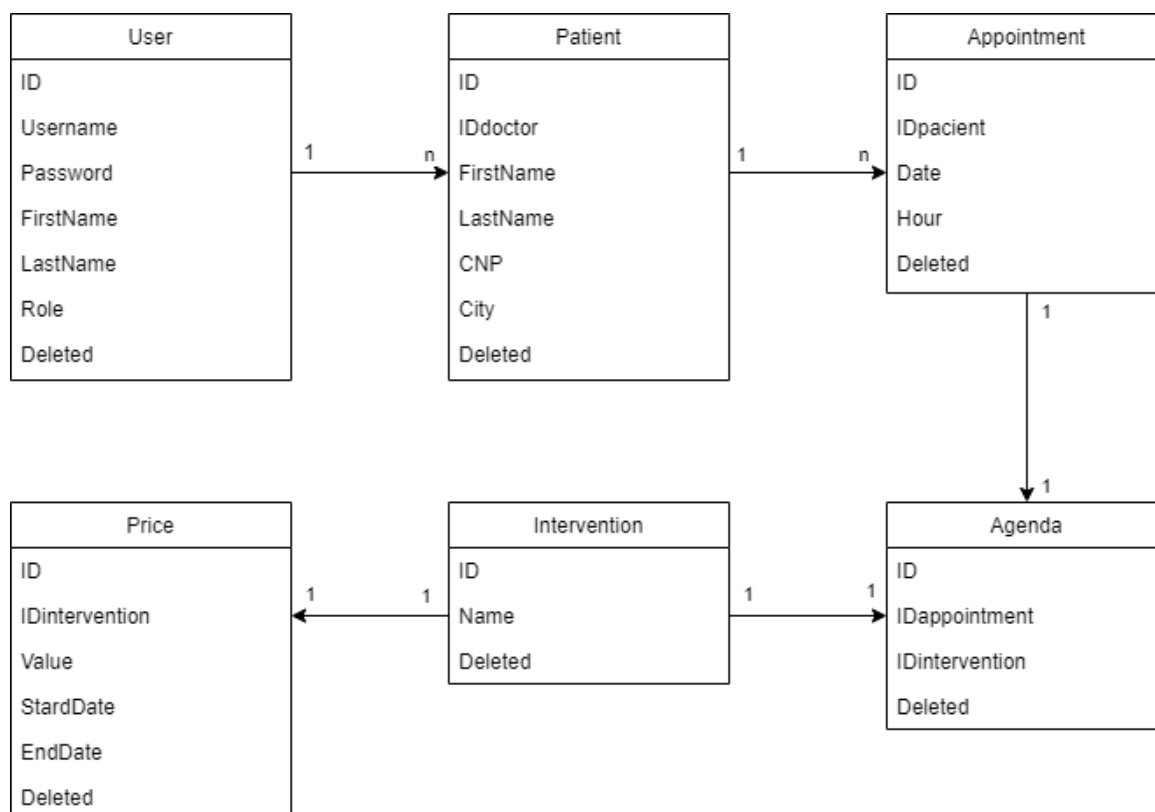
/***** Script for SelectTopNRows command
SELECT TOP (1000) [ID]
      , [IDappointment]
      , [IDintervention]
      , [Deleted]
FROM [dentalOfficeDB].[dbo].[Agenda]
  
```

100 %

Results Messages

	ID	IDappointment	IDintervention	Deleted
1	1	1	3	yes
2	2	2	3	yes
3	3	3	4	yes
4	4	4	4	yes
5	5	5	6	yes
6	6	6	4	no
7	7	6	6	no
8	8	7	6	no

## Relația tabelelor



## Relațiile de date

### 1. Utilizatori:

Tabelul de utilizatori are cheia primară (Pk) și este un ID de utilizator unic. De asemenea, are și o cheie străină (Fk), IDdoctor, care face referire la tabelul de înregistrare a pacienților.

### 2. Pacienți:

Tabelul de pacienți are o cheie primară (Pk) care este ID-ul pacientului și una străină (Fk), IDpacient mai exact care face referire la tabela de programări.

### 3. Intervenții:

Tabelul de intervenții are propria cheie primară (PK) care este ID-ul intervenției dar are și o cheie străină (Fk), IDintervention, care face referire la tabelele de preț și agendă.

#### **4. Programări:**

Tabelul de programări are o cheie primară (PK) care este ID-ul programării. De asemenea are și cheie străină, IDappointment mai exact, care face referire la tabela agenda.

#### **5. Preț:**

Tabelul de prețuri are o cheie primară (PK) care este ID-ul prețului.

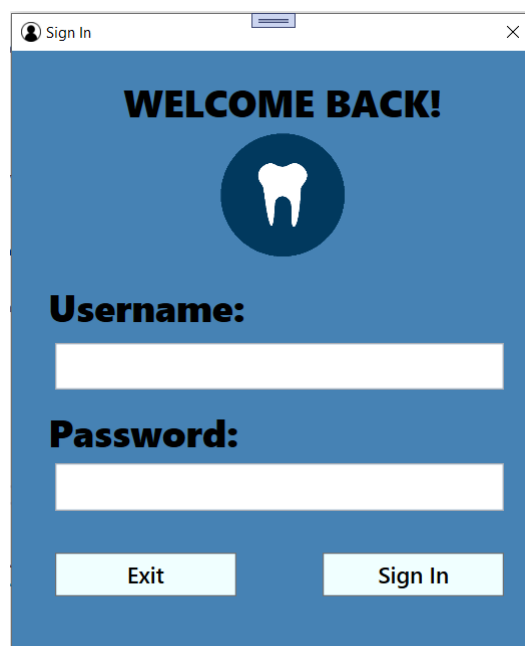
#### **6. Agenda:**

Tabela agenda are propria cheie primară (PK) care este ID-ul agendei.

### **3.2 User Interface Design**

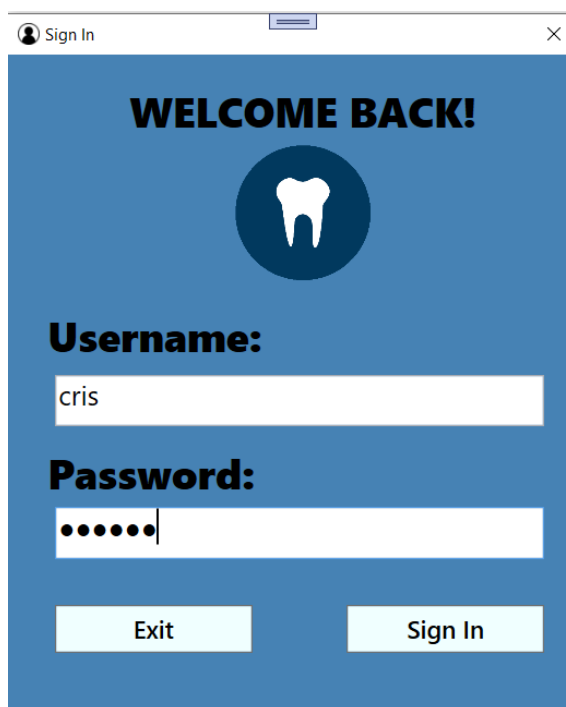
Aplicația este de tip desktop și poate funcționa pe majoritatea sistemelor de operare, de tip Windows.

Odată instalată, la deschidere apare o fereastră de logare. Aceasta solicită introducerea unor date de utilizator, respectiv un nume de utilizator și o parolă, ce pot corespunde unui medic sau unui administrator.



*Fig.1 Interfața de logare*

După ce s-au introdus datele, corecte, aplicația identifică dacă acestea corespund unui medic sau unui administrator și continuă cu meniul principal corespunzător acestuia.



*Fig. 2 Interfața de logare cu date introduse*

Meniul principal al administratorului, prezintă un mesaj de primire, cu numele utilizatorului, o listă cu toți doctorii înregistrați în baza de date, precum și una cu serviciile oferite de cabinetul stomatologic. Administratorul are acces la butoane pentru ștergerea sau adăugarea doctorilor, dar și a serviciilor. În plus, pentru servicii se pot verifica prețurile, cât și modifica valoarea acestora.

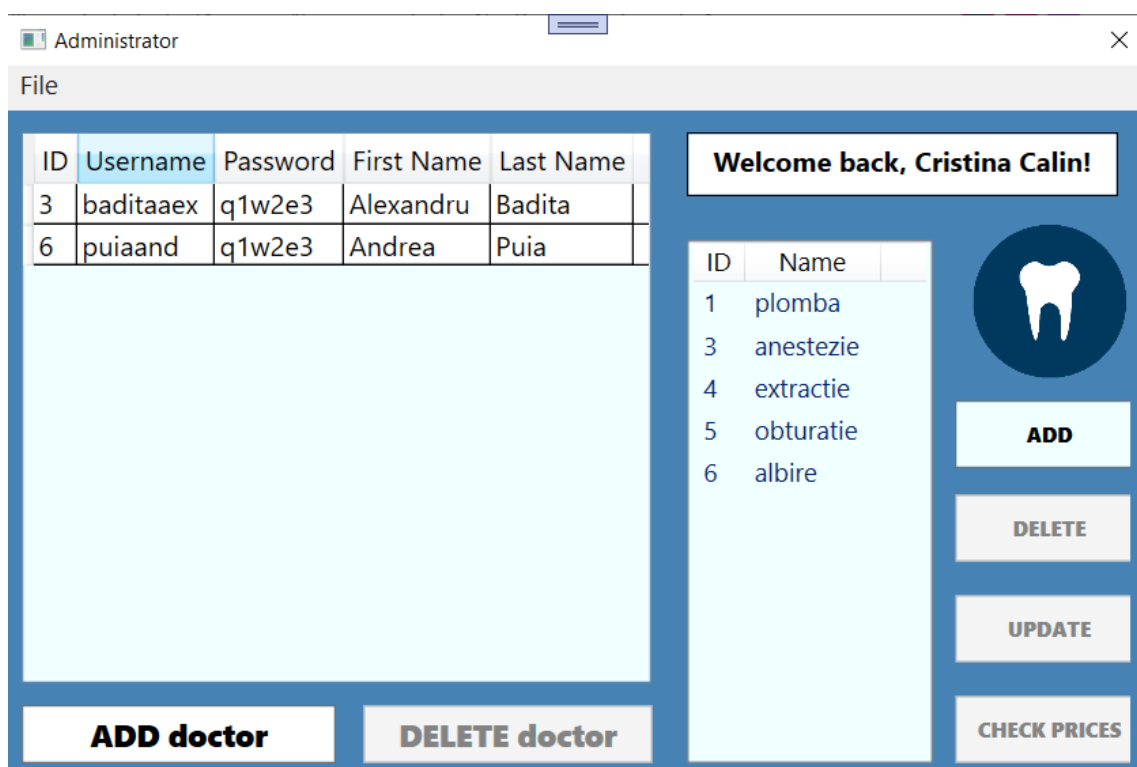


Fig. 3 Meniul principal al administratorului

În partea dreaptă a ferestrei se află serviciile oferite, iar cele patru butoane deservește acestora. Prin apăsarea butonului "ADD", se va deschide o nouă fereastră, ce permite adăugarea unui nou serviciu în baza de date. Aceasta, după cum se poate observa și în figura de mai jos, are nevoie de trei informații, respectiv numele, prețul și data de la care va fi disponibil acest serviciu. De asemenea, prezintă și două butoane, respectiv "BACK", pentru a anula introducerea acestor informații în baza de date, și "ADD", pentru a confirma adăugarea acestui serviciu.

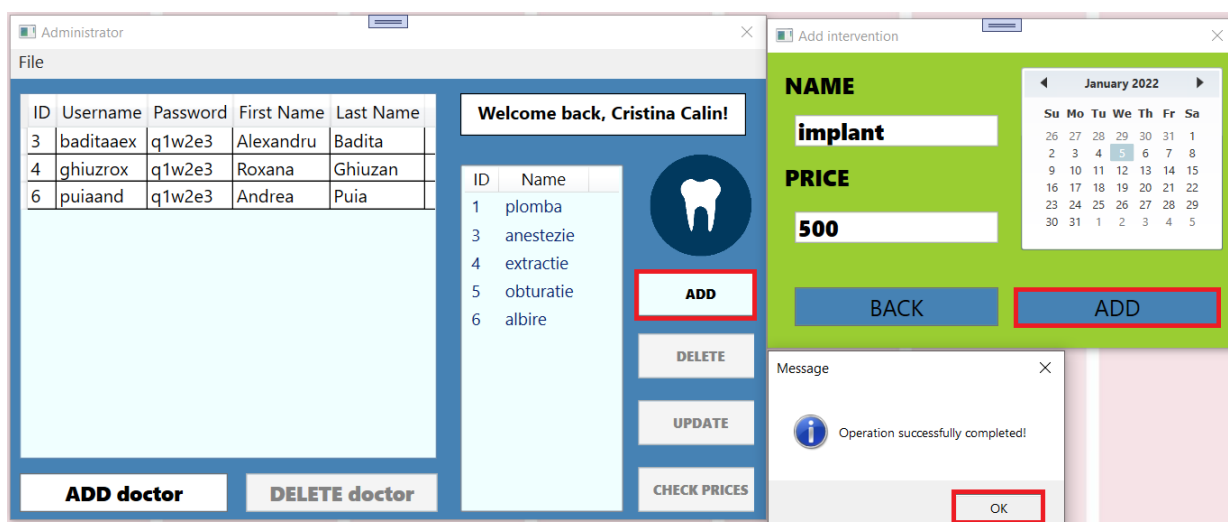


Fig.4 Meniul de adăugare al intervențiilor

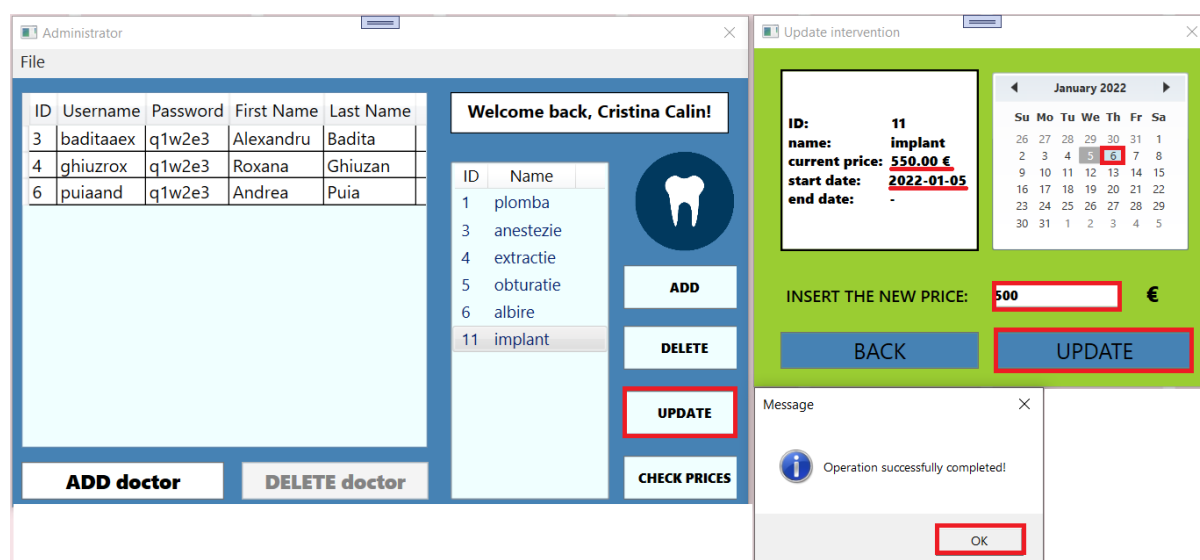


Fig.5 Meniu de actualizare al intervențiilor

După cum se poate observa în Figura 3, pe lângă butonul de “ADD”, mai există cel de “DELETE”, “UPDATE” și “CHECK PRICES”. Acestea sunt dezactivate, până la selectarea unui serviciu, după care permit ștergerea, modificarea și respectiv vizualizarea acestuia.

Butonul de “UPDATE” v-a deschide o fereastră nouă, unde se poate observa prețul curent și data de la care s-a stabilit acesta, iar mai apoi se poate introduce un nou preț și data de la care v-a fi valabil aceste.

Butonul de “CHECK PRICES” deschide o nouă fereastră, unde se pot verifica toate prețurile care au fost stabilite pentru o intervenție, de la o dată de început, până în prezent sau chiar și prețuri care urmează să fie valabile. Butonul de “BACK” permite navigarea înapoi la meniul principal.

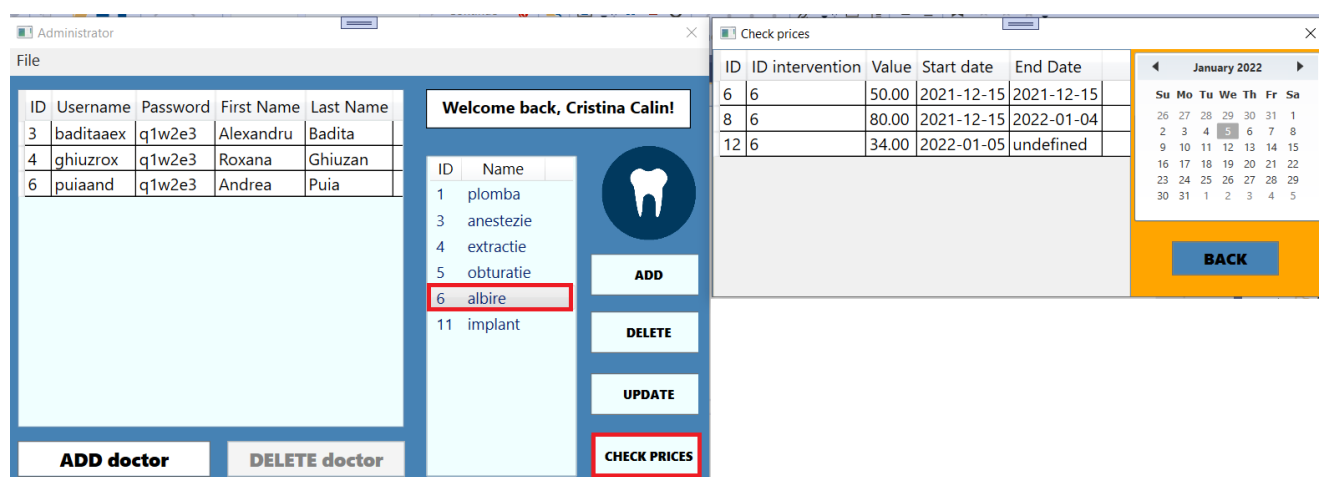


Fig.6 Fereastra de verificare a prețurilor unei intervenții

În partea stângă, a meniului principal, se află informațiile despre doctori. Butonul de “ADD doctor” deschide un formular pentru introducerea datelor unui nou doctor, respectiv nume de utilizator, prenume, nume de familie, parola, confirmarea parolei, după care există opțiunea de “BACK”, pentru anularea operațiunii, și cel de “REGISTER” pentru confirmarea datelor introduse.

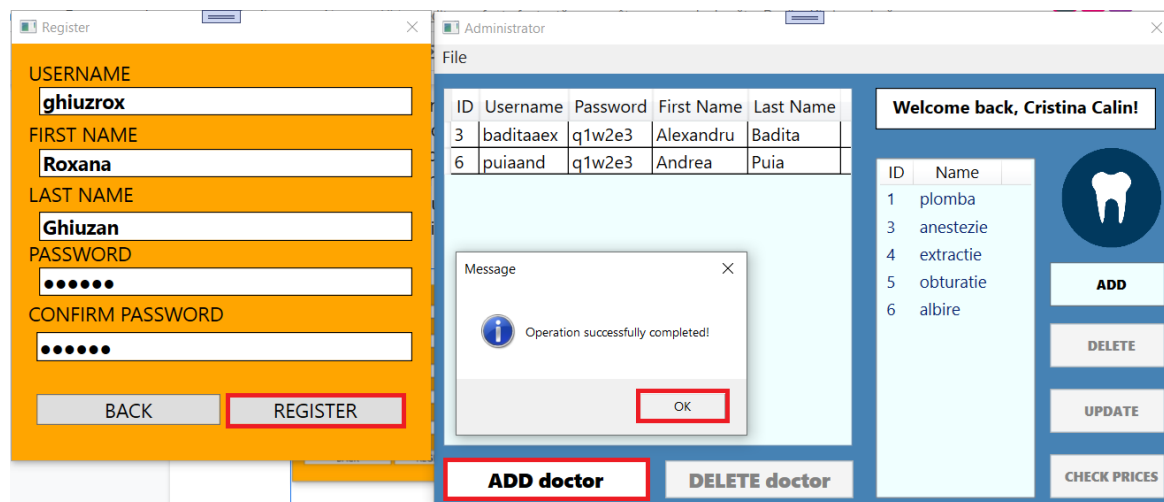


Fig.7 Meniul de înregistrare al doctorilor

Butonul de “DELETE doctor” este dezactivat, până la selectarea unui doctor din lista prezentată, după care acesta permite ștergerea doctorului respectiv din baza de date.

De asemenea, în partea stângă sus, există butonul dropdown “File”. Acesta ne oferă trei opțiuni: “Gains”, pentru a verifica încasările lunare ale unui medic, “Occupancy”, pentru a verifica disponibilitatea unui medic într-o anumită perioadă, și “Sign out”, pentru delogare.

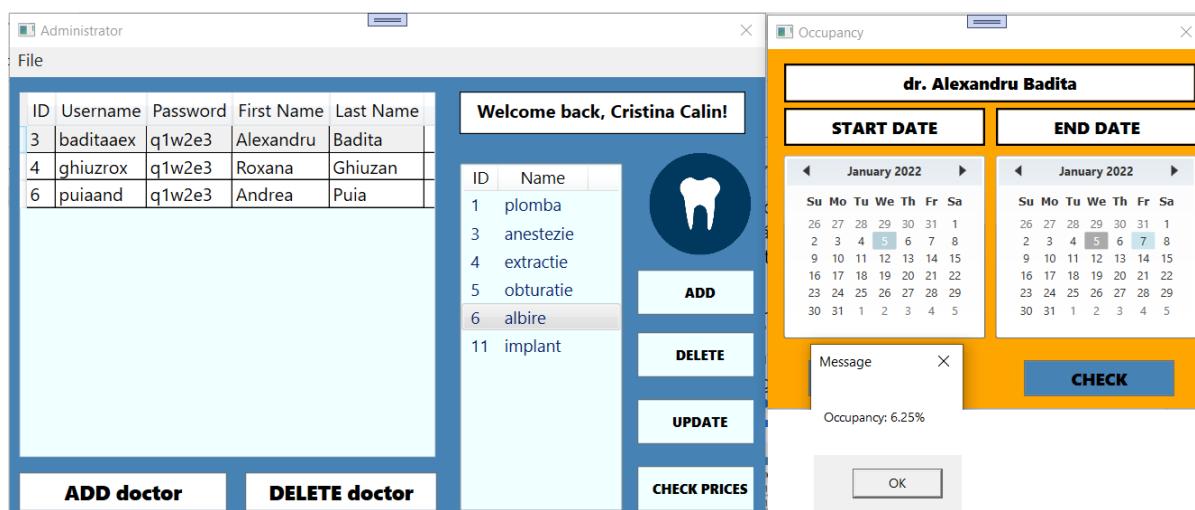


Fig.8 Meniul de disponibilitate al doctorilor

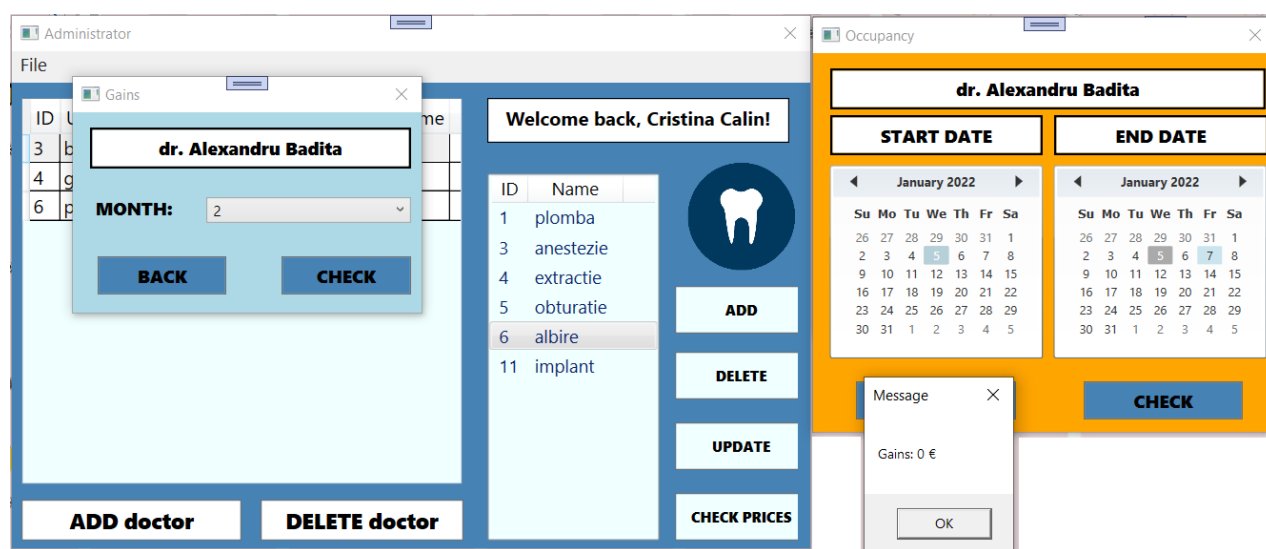


Fig.9 Meniul de calculare al încasărilor unui doctor

Meniul principal al medicului, prezintă un mesaj de primire, cu numele utilizatorului, o listă cu toți pacienții ai acestuia, o listă cu programări, precum și una cu serviciile oferite de cabinetul stomatologic.

În partea de sus a meniului se află lista cu pacienți, iar în dreapta acesteia butoanele de “ADD patient” și “DELETE patient”, acesta este dezactivat, până la selectarea unui pacient din listă, după care efectuează ștergerea acestuia.

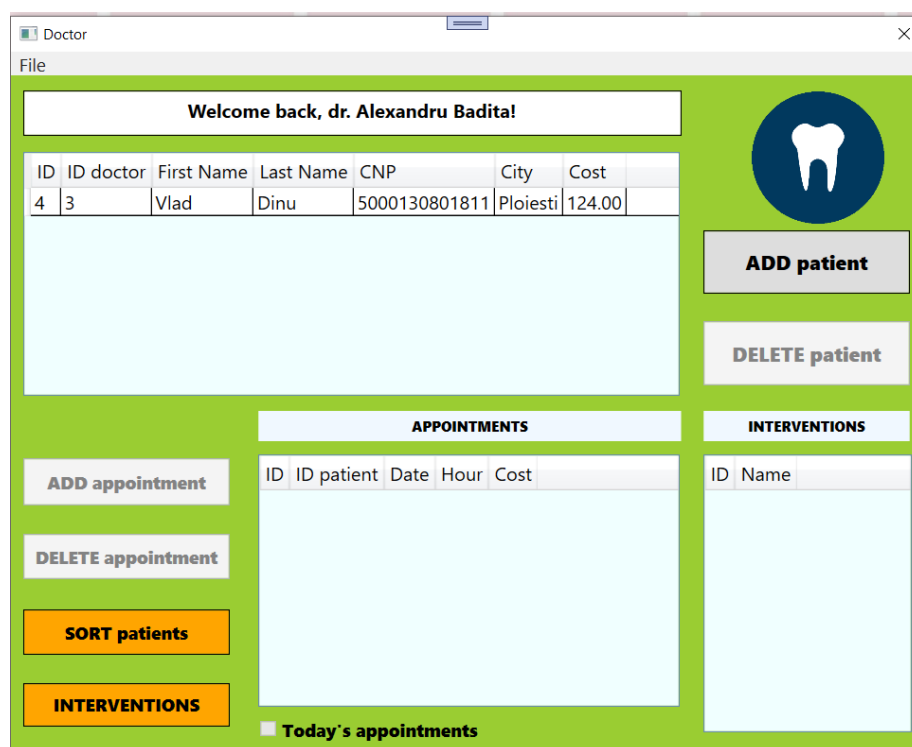


Fig.10 Meniul principal al doctorului



Prin apăsarea butonului “ADD patient” se deschide un formular pentru introducerea datelor acestuia, respectiv prenume, nume, cnp, oraș. După care se poate confirma introducerea acestora în baza de date prin apăsarea butonului de “ADD” sau anula prin apăsarea butonului de “BACK”.

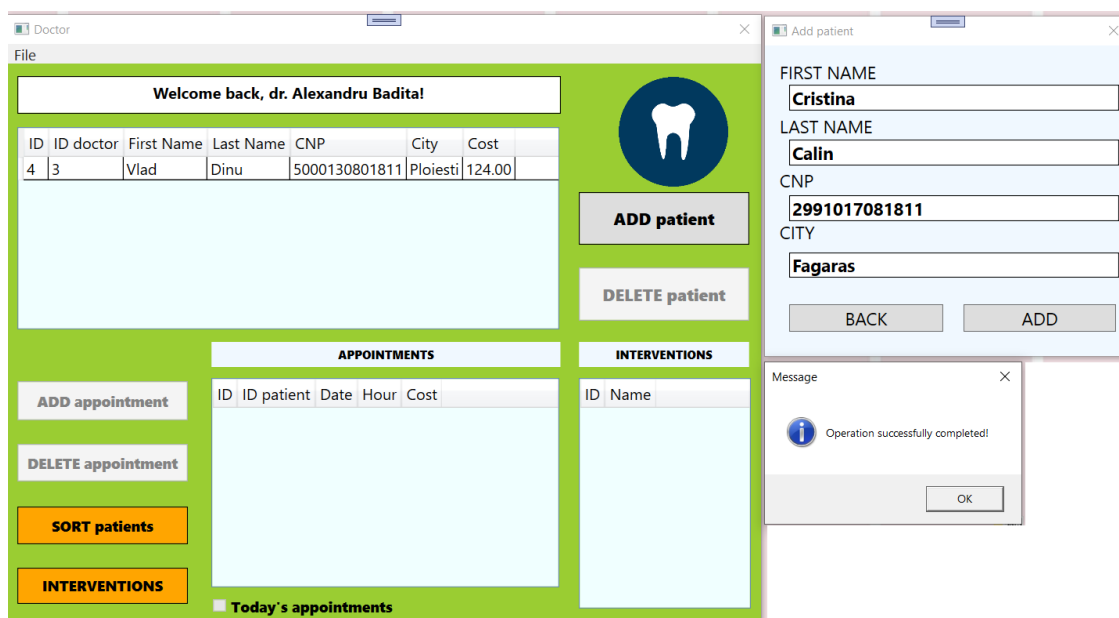


Fig.11 Meniul de adăugare a pacienților

În partea de jos a ecranului se află lista de programări. Prin selectarea unui pacient, se activează butonul de “ADD appointment”, și se deschide o nouă fereastră, unde se selectează ce intervenții vor fi necesare, data și ora la care v-a avea loc programarea. După care se confirmă datele introduse prin apăsarea butonului de “CONFIRM” sau se anulează cu butonul de “BACK”.

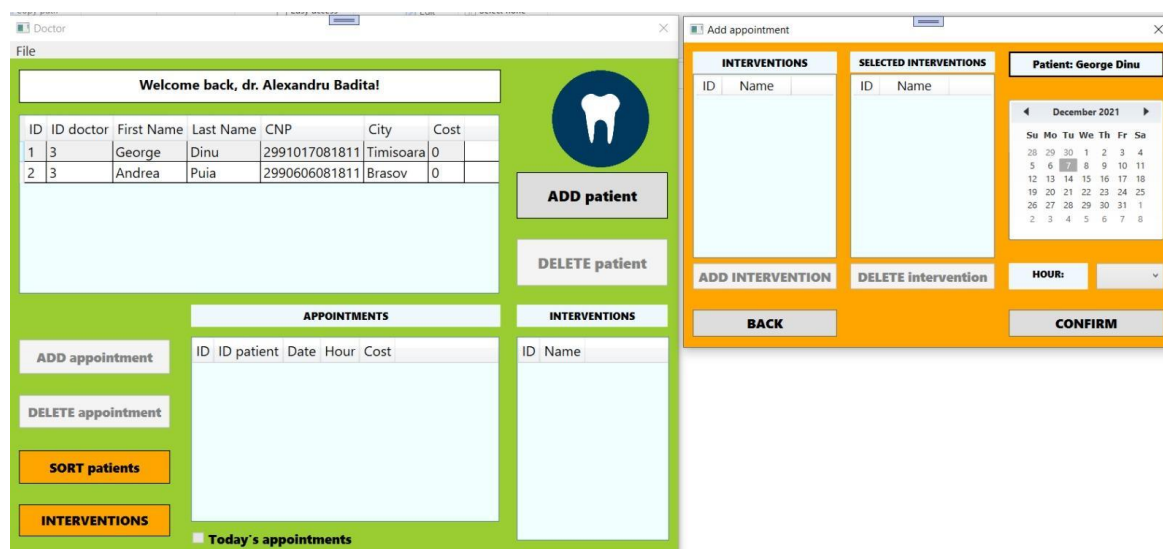


Fig.12 Meniul de creare a programărilor

De asemenea, se poate selecta o programare din lista prezentată, iar prin apăsarea butonului “DELETE appointment” se v-a șterge, se pot vizualiza programările doar pentru ziua curentă, prin bifarea opțiunii “Today’s appointments”, lista de pacienți folosind butonul “Patients” sau vizualizarea listei de intervenții disponibile cu butonul de “INTERVENTIONS”.

## **Capitol 4**

### **Implementare**

#### **4.1 Mod de funcționare**

Aplicația este de tip desktop și poate funcționa pe majoritatea sistemelor de operare.

Odată instalată, la deschidere apare o fereastră de logare. Aceasta solicită introducerea unor date de utilizator (de medic sau de administrator).

După ce s-au introdus datele, corecte, aplicația identifică dacă acestea corespund unui medic sau unui administrator și continuă cu meniul principal corespunzător acestuia.

Cel al stomatologului, prezintă o listă de programări, o listă de servicii și prețurile aferente acestora. Fiecare programare sau serviciu poate fi editat sau șters, dar există și posibilitatea de a crea noi instanțe ale acestora.

Meniul administratorului, permite acestuia să gestioneze conturile stomatologilor și să modifice sau șteargă intervenții.

La final, fiecare utilizator are opțiunea de a se deloga din aplicație, sau v-a fi delogat automat în momentul încheierii acesteia.

#### **4.2 Mod de realizare**

Mediul de dezvoltare al aplicației este Microsoft Visual Studio. Este un mediu de dezvoltare integrat (IDE) de la Microsoft. Este folosit pentru dezvoltarea de programe pe calculator, precum și a site-urile, aplicațiilor web, mobile și a serviciilor web.

Aplicația a fost realizată în C#, mai exact în .Net Framework. Acesta este un framework software, dezvoltat de Microsoft, ce oferă interfață pentru utilizator, acces la date, conectivitate la baza de date, comunicații în rețea, precum și alte funcționalități.

Ca bază de date, folosim SQL Server Express. Acesta este o versiune de bază de date SQL Server, oferită de Microsoft, ce servește aplicațiilor încorporate, la scară mică.

## **Capitol 5**

### **Testare**

Testarea software-ului poate fi declarată ca fiind procesul de validare și verificare a unui software/program/aplicație.

1. testăm dacă îndeplinește cerințele care i-au ghidat proiectarea și dezvoltarea
2. verificăm dacă funcționează conform așteptărilor

#### **5.1 Unit testing**

*Unit Testing* presupune testarea fiecărei unități sau a unei componente individuale a aplicației software. Este primul nivel de testare funcțională. Scopul din spatele testării unitare este de a valida componentele unității cu performanța acesteia.

O unitate este o singură parte testabilă a unui sistem software și testată în timpul fazei de dezvoltare a aplicației software.

Scopul testării unitare este de a testa corectitudinea codului izolat. O componentă de unitate este o funcție sau un cod individual al aplicației.

#### **Beneficii Unit Testing**

1. Remediază erorile la începutul etapei de dezvoltare. Acest lucru economisește mult timp, efort și costuri.
2. Testele unitare valide folosesc la realizarea unei documentații detaliate.

3. Ușurează modificarea și întreținerea codului.
4. De asemenea, impune modularitatea. Testele unitare sunt executate pe componente individuale.

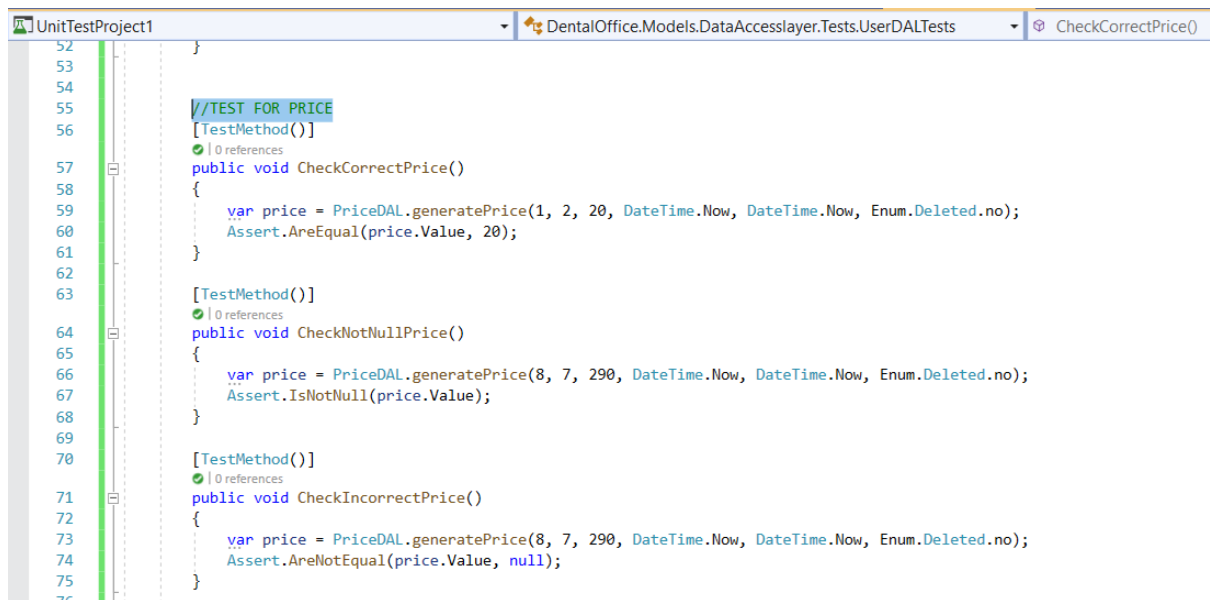
## 5.2 Exemple Unit Test Case

### 1.Test Case pentru clasa UserDAL



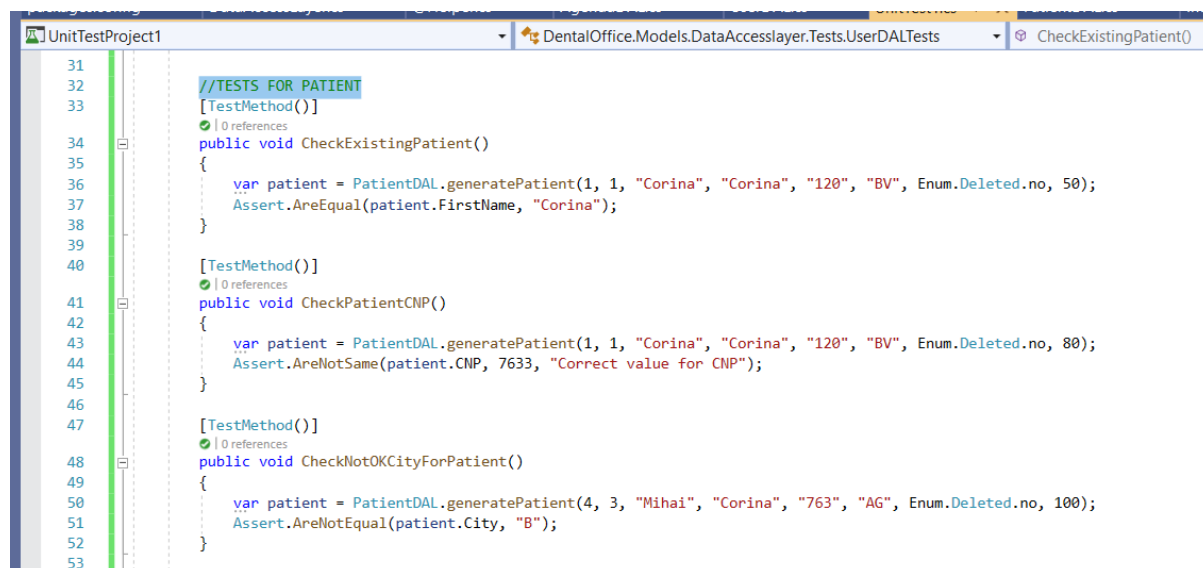
```
UnitTestProject1 | DentalOffice.Models.DataAccesslayer.Tests.UserDALTests | CheckUserName()
7 | 0 references
8 | public class UserDALTests
9 | {
10 |     //TESTS FOR USER
11 |     [TestMethod()]
12 |     public void CheckUserName()
13 |     {
14 |         var user = UserDAL.generateUser(1, "Andrea", "Parola", "Andrea", "Puia", Role.doctor, Enum.Deleted.no);
15 |         Assert.AreEqual(user.Username, "Andrea");
16 |     }
17 |
18 |     [TestMethod()]
19 |     public void CheckUserPassword()
20 |     {
21 |         var user = UserDAL.generateUser(1, "Andrea", "q1w2e3", "Andrea", "Puia", Role.doctor, Enum.Deleted.no);
22 |         Assert.AreSame(user.Password, "q1w2e3", "Incorrect password");
23 |     }
24 |
25 |     [TestMethod()]
26 |     public void CheckFirstNameType()
27 |     {
28 |         var user = UserDAL.generateUser(1, "cris", "q1w2e3", "Cristina", "Calin", Role.administrator, Enum.Deleted.no);
29 |         Assert.IsInstanceOfType(user.FirstName, typeof(string));
30 |     }
31 | }
```

### 2.Test Case pentru clasa PriceDAL

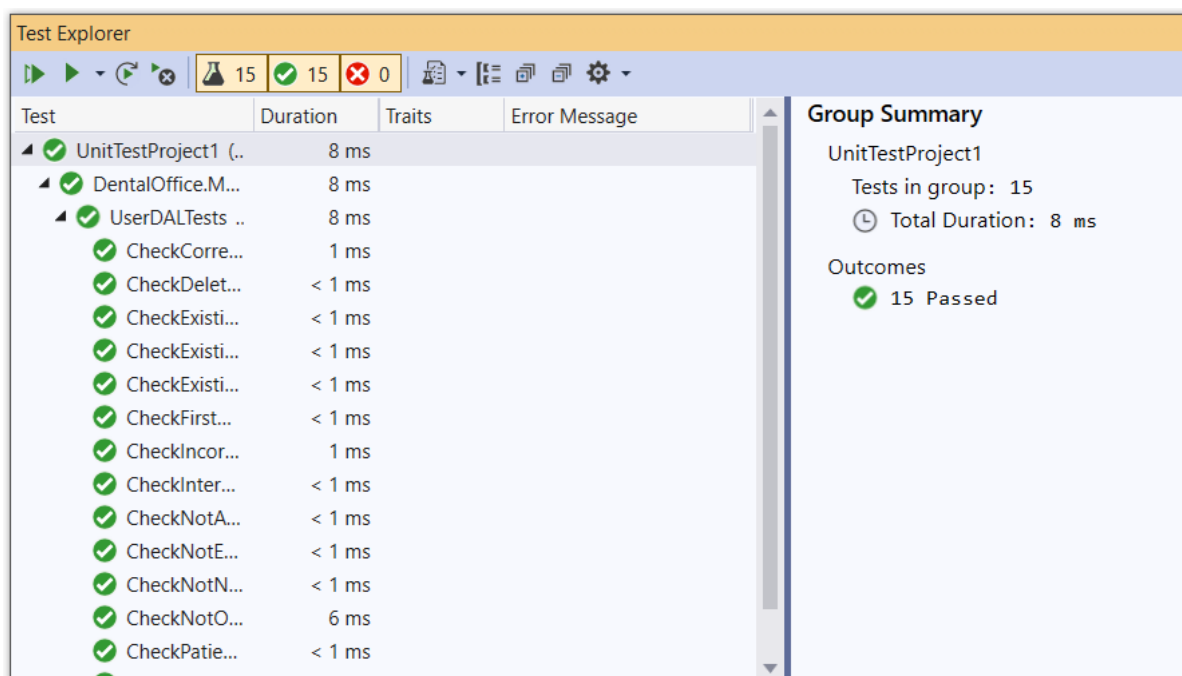


```
UnitTestProject1 | DentalOffice.Models.DataAccesslayer.Tests.UserDALTests | CheckCorrectPrice()
52 | }
53 |
54 | //TEST FOR PRICE
55 | [TestMethod()]
56 | public void CheckCorrectPrice()
57 | {
58 |     var price = PriceDAL.generatePrice(1, 2, 20, DateTime.Now, DateTime.Now, Enum.Deleted.no);
59 |     Assert.AreEqual(price.Value, 20);
60 | }
61 |
62 | [TestMethod()]
63 | public void CheckNotNullPrice()
64 | {
65 |     var price = PriceDAL.generatePrice(8, 7, 290, DateTime.Now, DateTime.Now, Enum.Deleted.no);
66 |     Assert.IsNotNull(price.Value);
67 | }
68 |
69 | [TestMethod()]
70 | public void CheckIncorrectPrice()
71 | {
72 |     var price = PriceDAL.generatePrice(8, 7, 290, DateTime.Now, DateTime.Now, Enum.Deleted.no);
73 |     Assert.AreNotEqual(price.Value, null);
74 | }
75 |
76 | }
```

### 3.Test case pentru clasa PatientDAL

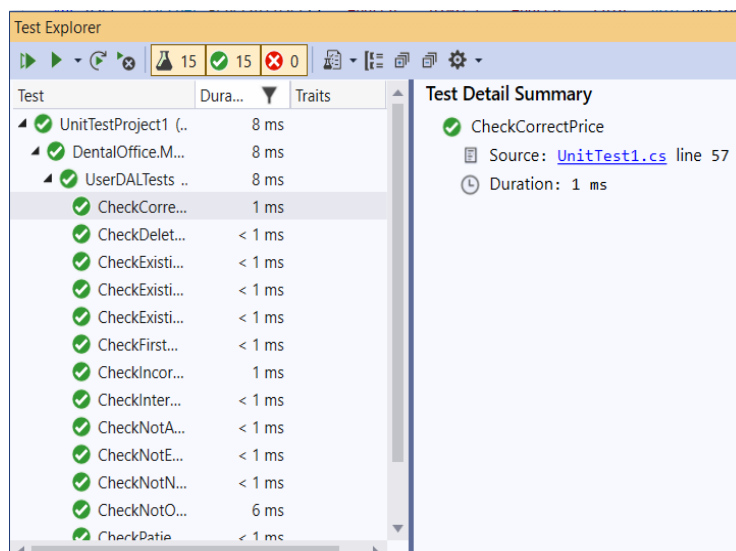


### Rularea testelor



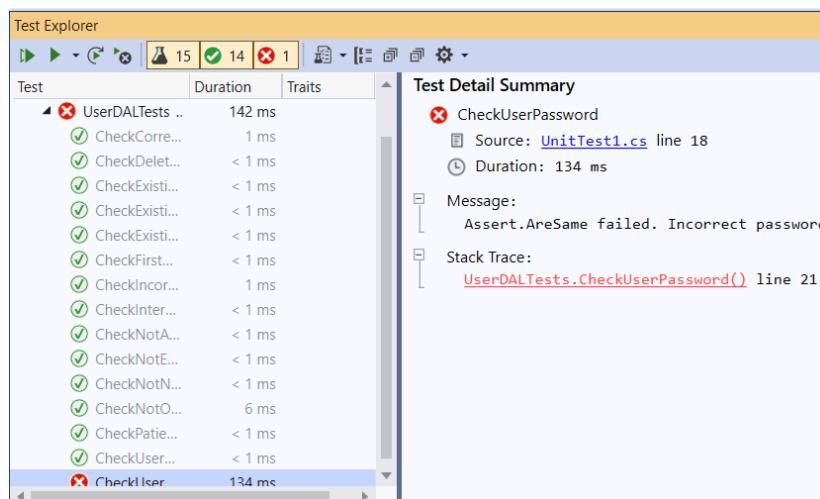
Imaginea este reprezentativă pentru rezultatul obținut în urma rulării testelor . Așadar, se observă că cele 15 teste efectuate pentru aplicație trec în timp util.

Pentru a vedea detaliile unui test individual, selectați testul pentru a se afișa informațiile aferente:



Panoul de detalii test afișează următoarele informații:

- Numele fișierului sursă și numărul rândului metodei de testare.
- Starea testului.
- Timpul scurs pentru a rula metoda de testare.



Dacă testul eșuează, se afișează și panoul de detalii:

- Mesajul returnat de cadrul testului unitar pentru test.
- Urmărirea stivei în momentul în care testul a eșuat.