Proiect Medii de proiectare - 2022

Baza de date:

CodUrgenta | Internari | Pacienti | Sectii | Utilizatori

Field Name	Data Type	Description (Optional)
IdCodUrgenta	AutoNumber	
DCodUrgenta	Short Text	
TotalIntrari	Number	

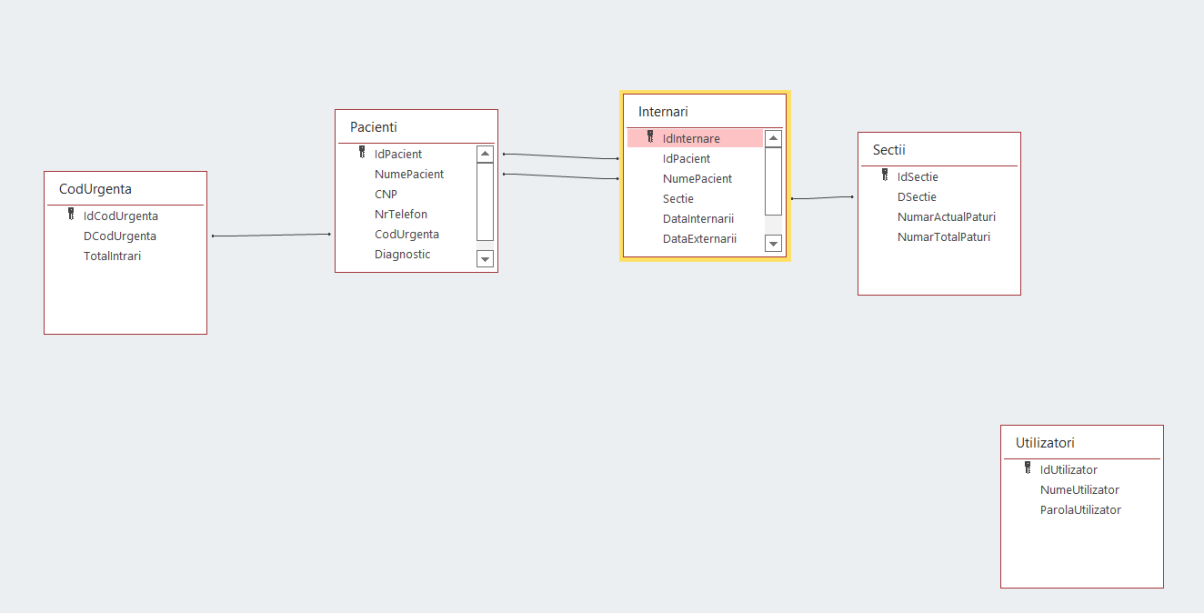
Field Name	Data Type	Description (Optional)
IdInternare	AutoNumber	
IdPacient	Short Text	
NumePacient	Short Text	
Sectie	Short Text	
DataInternarii	Date/Time	
DataExternarii	Date/Time	
Epicriza	Short Text	
PozaCard	Short Text	

Field Name	Data Type	Description (Optional)
IdPacient	AutoNumber	
NumePacient	Short Text	
CNP	Number	
NrTelefon	Number	
CodUrgenta	Short Text	
Diagnostic	Short Text	
Decizie	Short Text	

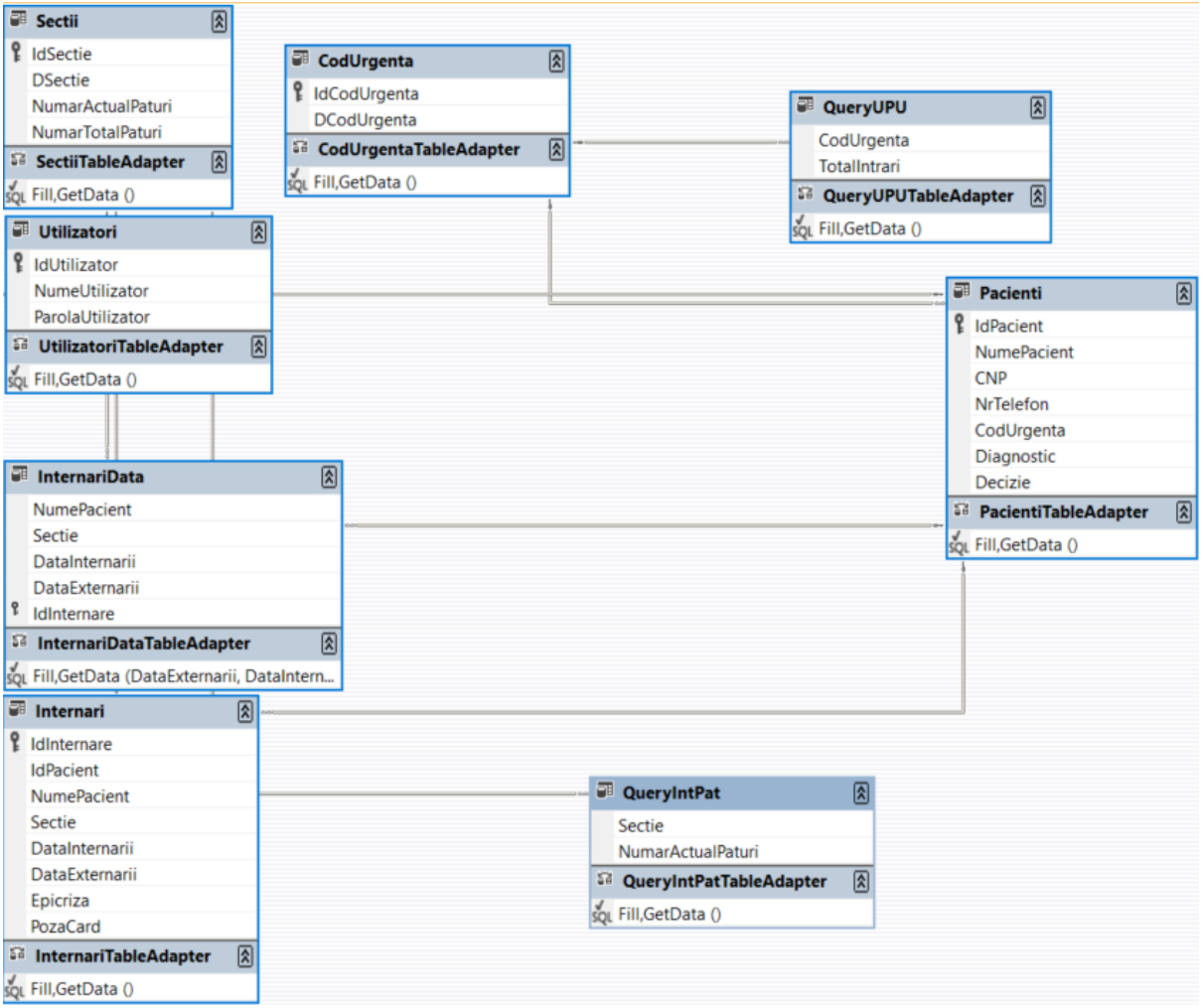
Field Name	Data Type	Description (Optional)
IdSectie	AutoNumber	
DSectie	Short Text	
NumarActualPaturi	Number	
NumarTotalPaturi	Number	

Field Name	Data Type	Description (Optional)
IdUtilizator	AutoNumber	
NumeUtilizator	Short Text	
ParolaUtilizator	Short Text	

Realizat de Olaru Andrei(Grupa 3)



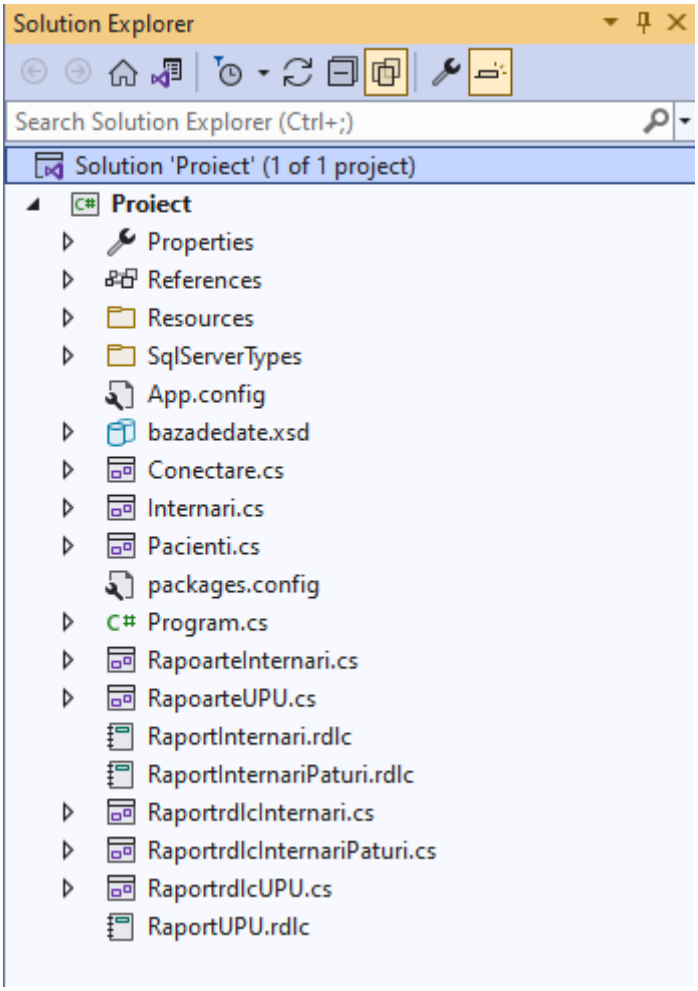
DataSet Visual Studio:



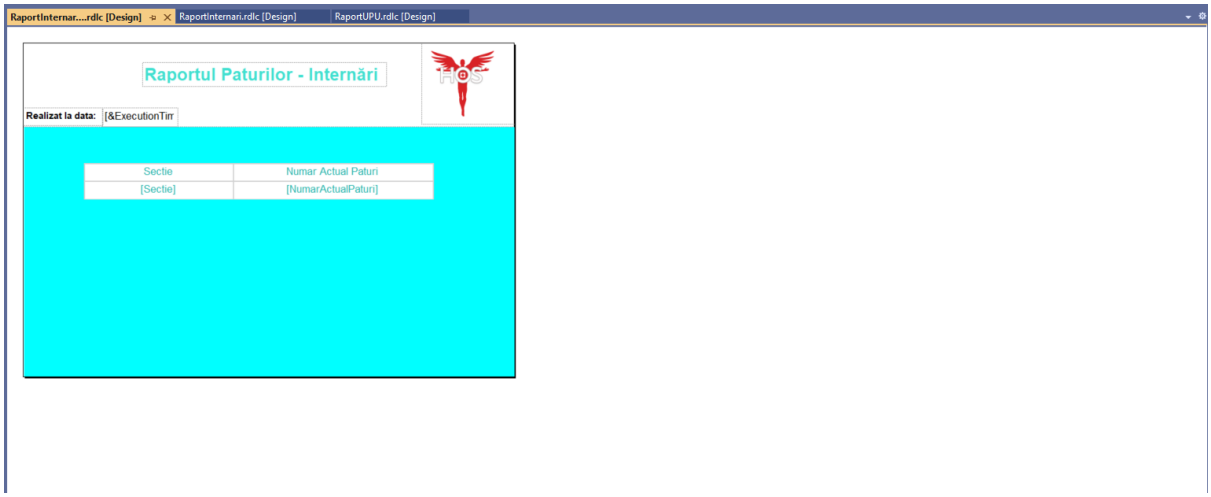
Realizat de Olaru Andrei(Grupa 3)

Proiect Medii de proiectare - 2022

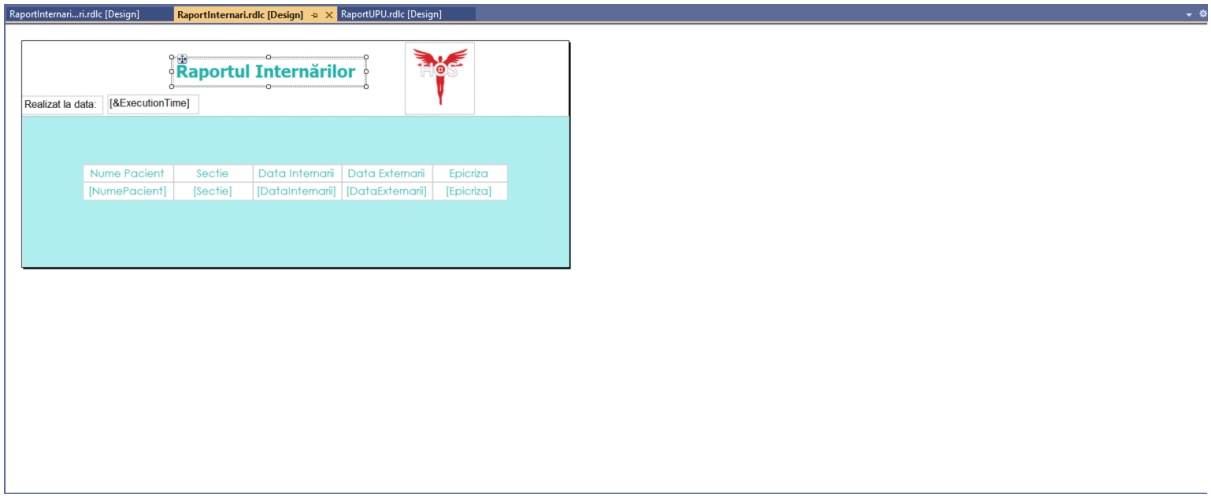
Solution Explorer:

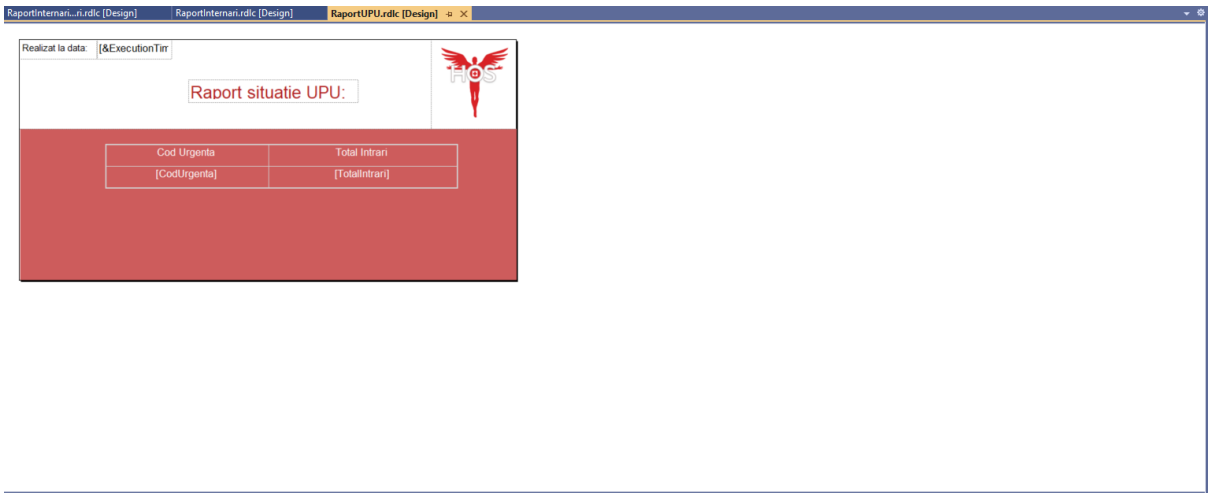


Rapoarte RDLC(3):



Realizat de Olaru Andrei(Grupa 3)

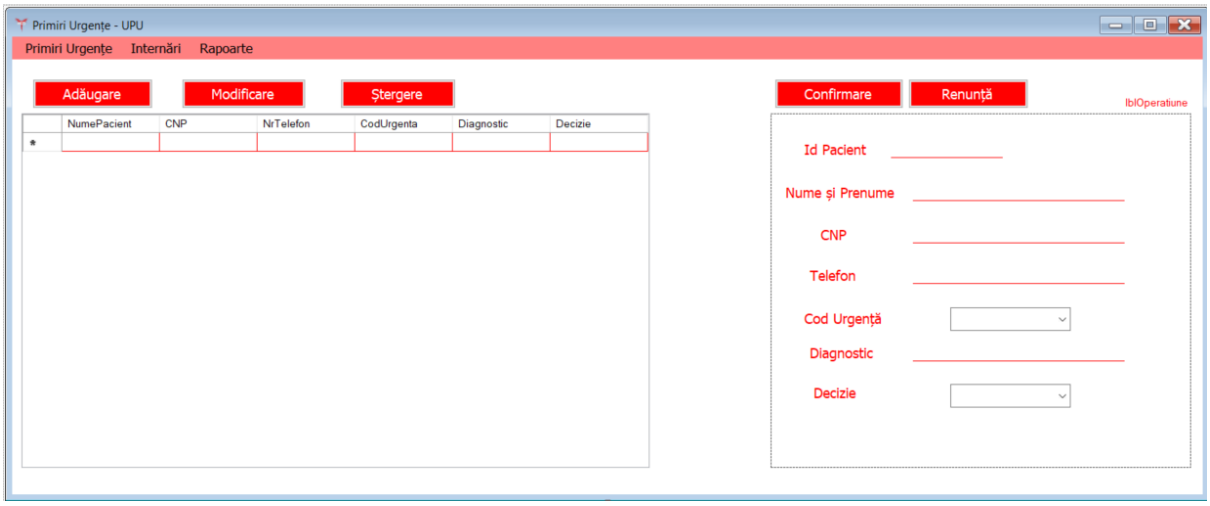




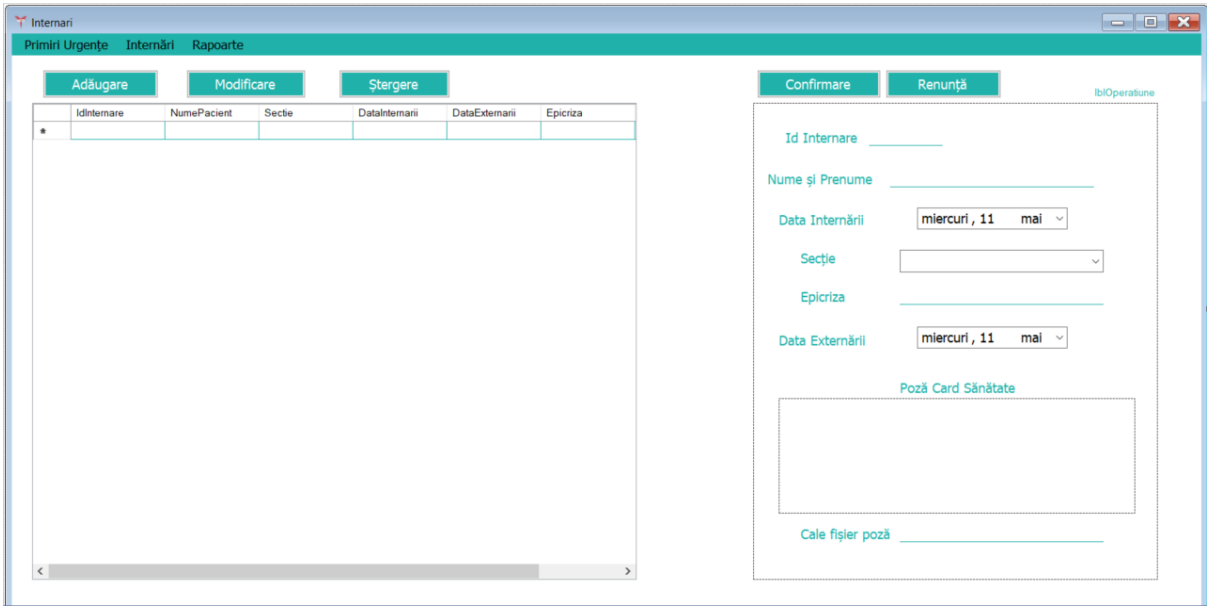
Form Conectare:
(Credențiale: admin/admin)

Proiect Medii de proiectare - 2022

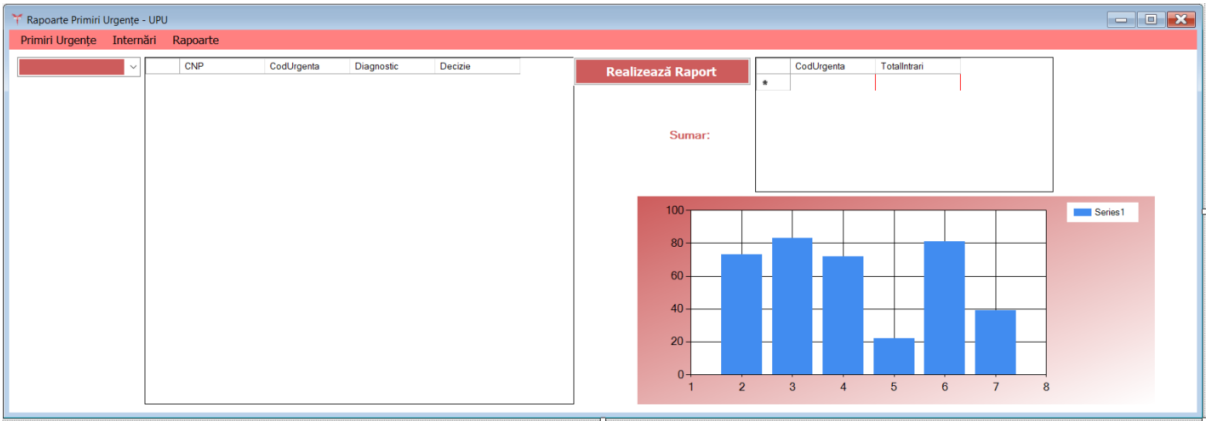
Form „Primiri Urgente – UPU”:



Form „Internari”:

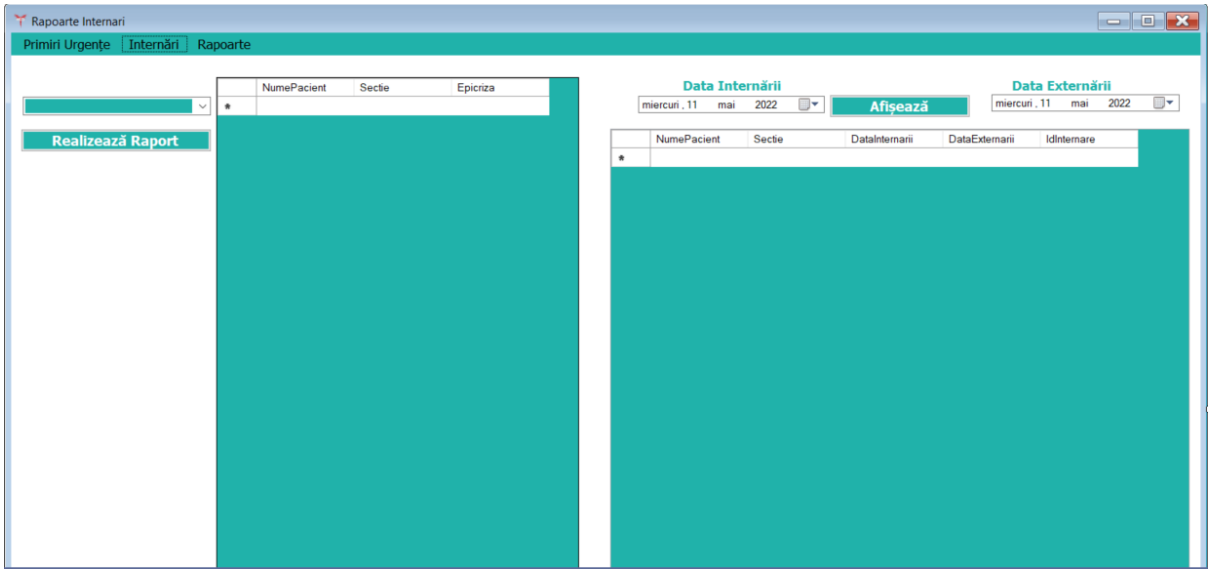


Form „Rapoarte UPU”:

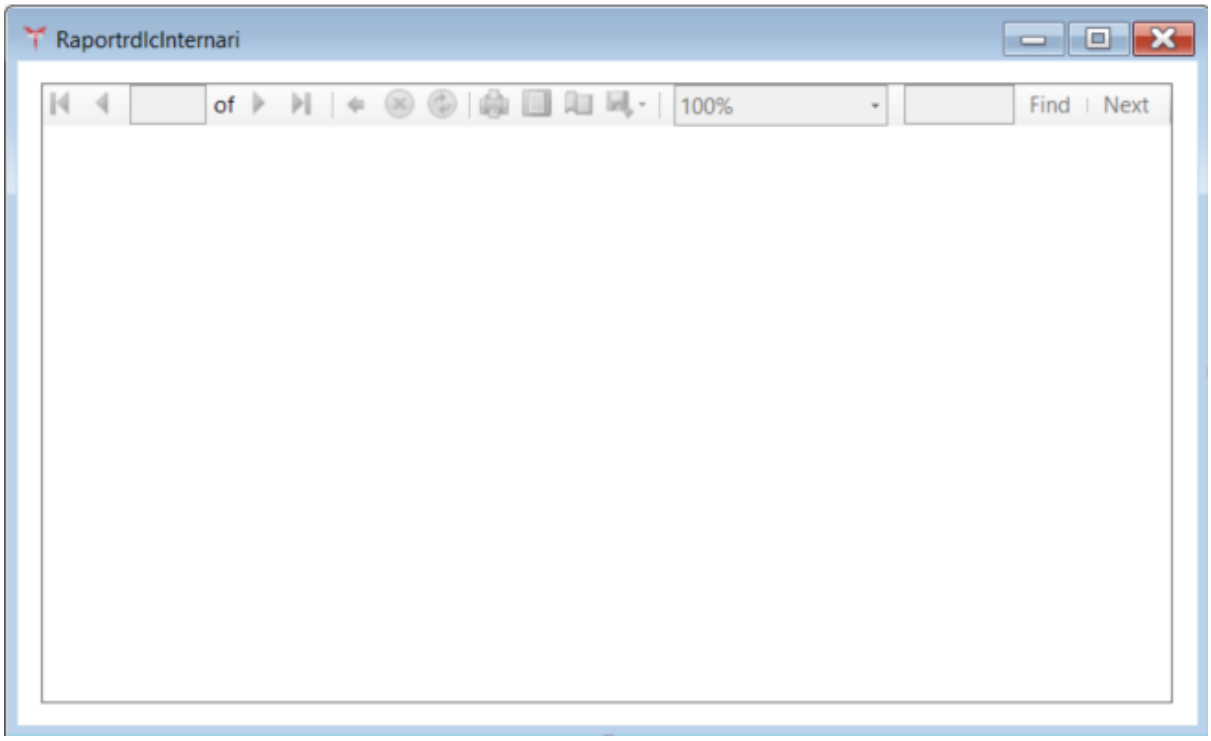


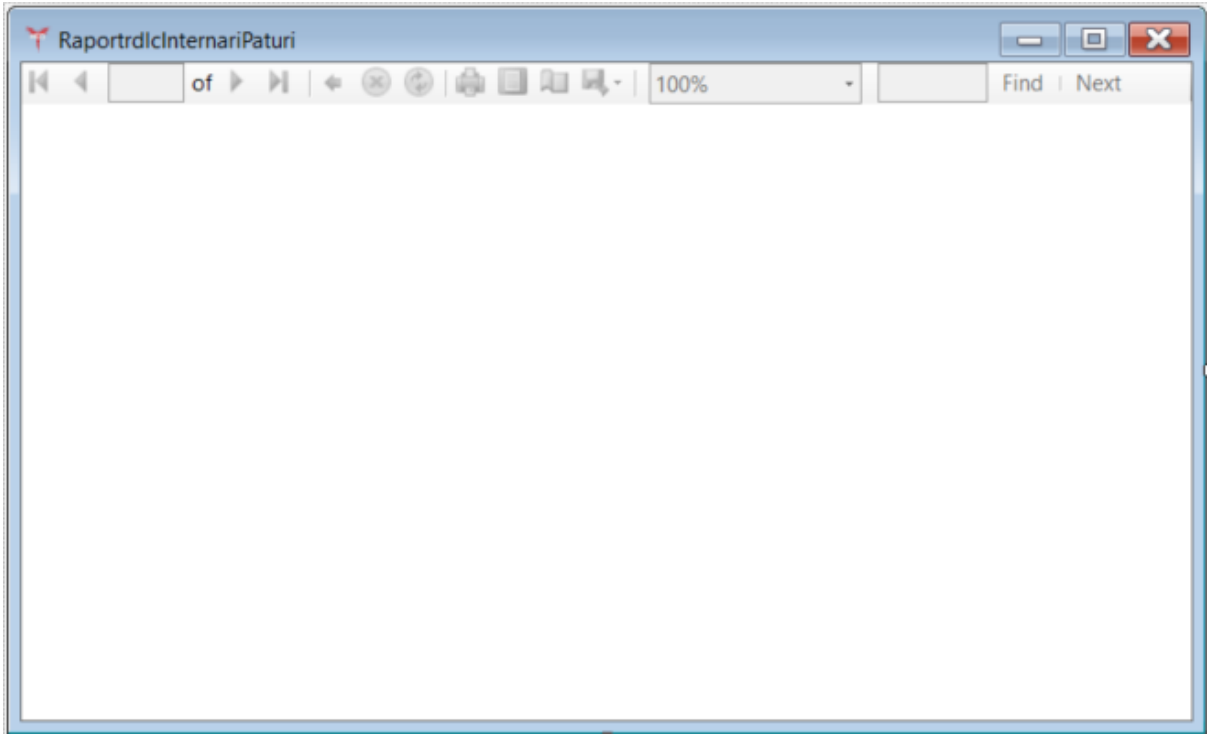
Realizat de Olaru Andrei(Grupa 3)

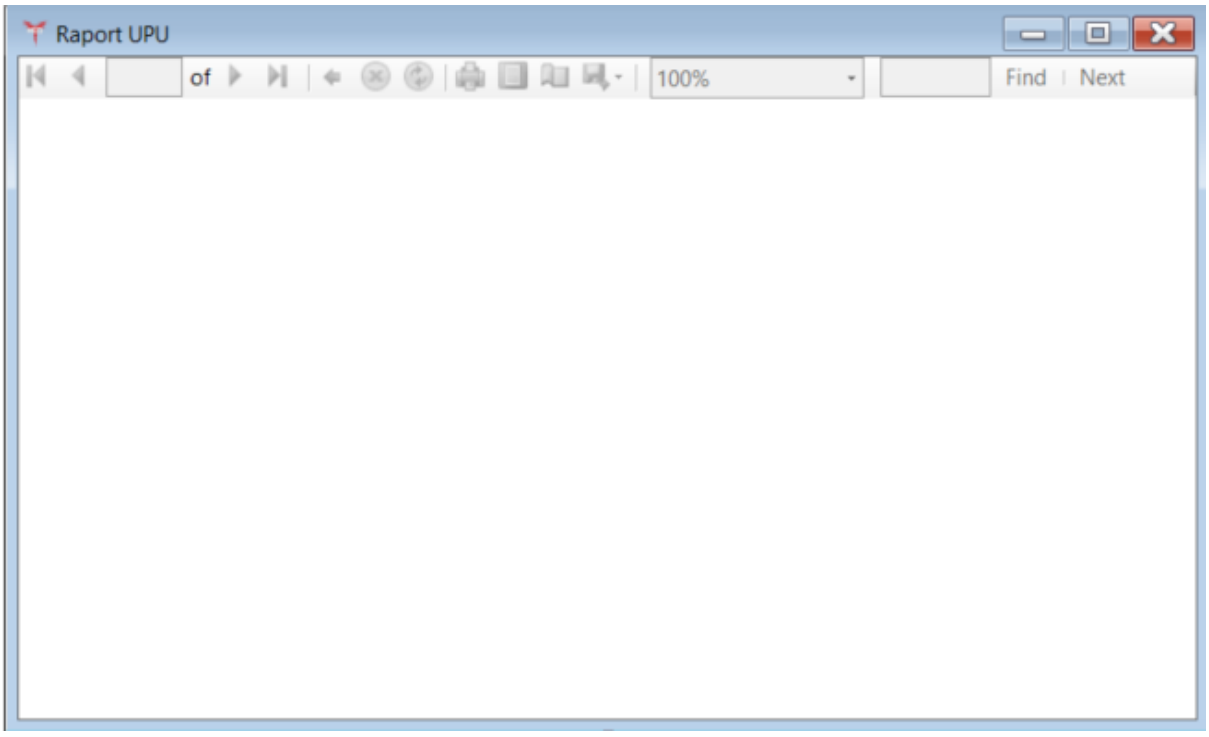
Form „Rapoarte Internari”:



Form-urile aferente Rapoartelor de tip RDLC:




RaportrdIcInternariPaturi
of
100%
Find
Next


Raport UPU
of
100%
Find
Next

COD:

Form conectare:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.OleDb;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Proiect{
    public partial class Conectare : Form{
        private OleDbConnection con = new
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\\Users\\olaru\\Desktop\\ProiectMPP2\\Resurse\\HOS2.mdb");
        private OleDbCommand cmd = new OleDbCommand();
        private OleDbDataReader rdr;
        public Conectare(){
            InitializeComponent();
        }

        private void btnConectare_Click(object sender, EventArgs e){
            if (con.State == ConnectionState.Closed){
                con.Open();
            }
            string conectare = "SELECT * FROM Utilizatori WHERE NumeUtilizator
='" + txtNumeUtilizator.Text + "' AND ParolaUtilizator ='" + txtParola.Text +
"'";
            cmd = new OleDbCommand(conectare, con);
            rdr = cmd.ExecuteReader();
            if (rdr.Read() == true)
            {
                Pacienti pacienti = new Pacienti();
                pacienti.Show();
                this.Hide();
            }
            else
            {
                MessageBox.Show("Credențialele introduse sunt incorecte, vă rugăm
să reîncercați!", "Conectare Eșuată!", MessageBoxButtons.OK,
MessageBoxIcon.Error);
                txtNumeUtilizator.Text = "";
                txtParola.Text = "";
                txtNumeUtilizator.Focus();
            }
        }
    }
}
```

Form Internari:

```
using System;
using System.Data;
using System.Data.OleDb;
using System.Windows.Forms;

namespace Proiect{
    public partial class Internari : Form{
        public Internari(){
            InitializeComponent();
        }
        private void Internari_Load(object sender, EventArgs e){
            this.sectiiTableAdapter.Fill(this.bazadedate.Sectii);
            A1();
            txtIdInternare.Enabled = false;
        }
        private void A1(){
            this.internariTableAdapter.Fill(this.bazadedate.Internari);
            //Configurez Poza cardului de sanatate;
            pozacardsanatate.SizeMode = PictureBoxSizeMode.StretchImage;
            //Initiez protectia Gridului;
            dataGridView1.AllowUserToAddRows = false;
            dataGridView1.AllowUserToDeleteRows = false;
            dataGridView1.ReadOnly = true;
            A3();
        }
        private void A2(){
            //Configurez butoanele;
            configureazaButoane(false);
            //Dezleg controalele de la Panelul Principal
            legareControale(false);
            //"Elimin" protectia controalelor din Panel;
            protectiePanel(false);
            //Modific lblOperatiune pentru adaugare;
            lblOperatiune.Text = "ADAUG";
            //Pun focusul pe primul camp de completare;
            txtNumePacient.Focus();
            //Pregatesc campurile pentru introducerea datelor;
            golireCampuri();
        }
        private void A3(){
            //Initializez label-ul Operatiune drept "null";
            lblOperatiune.Text = "";
            //Configurez butoanele;
            configureazaButoane(true);
            //"Pun" protectia asupra Panelului Principal;
            protectiePanel(true);
            //leg controalele;
            legareControale(true);
        }
        private void A4(){
            if (lblOperatiune.Text == "ADAUG"){
                if (!validareaCampurilor()) return;//conditia de a completa
campurile;
                adauga_internare();//Adaug internarea;
                golireCampuri();//Pregatesc campurile pentru urmatoarea;
                txtNumePacient.Focus();//Pun focus pe primul camp;
                refresh_grid(internariBindingSource.Position);//Am dat refresh la
grid, pentru a vizualiza in timp real adaugarea internarii;
            }
            else if (lblOperatiune.Text == "MODIFIC"){
```

```
                if (!validareaCampurilor()) return;//conditia de a completa
campurile;
                modific_internare();//Modific internarea;
                refresh_grid(internariBindingSource.Position);//Am dat refresh la
grid, pentru a vizualiza in timp real modificarea internarii;
                A3();
            }else{
                MessageBox.Show("Operatiunea a eșuat. Vă rugăm să încercați din
nou!");
            }
        }
        private void A6(){
            if (lblOperatiune.Text == "") return;
            if (openFileDialog1.ShowDialog() == DialogResult.OK){
                txtpozacard.Text = openFileDialog1.FileName;
                pozacardsanatate.ImageLocation = txtpozacard.Text;
            }
        }
        private void A7(){
            //Configurez butoanele;
            configureazaButoane(false);
            //Dezleg controalele de la Panel;
            legareControale(false);
            //"Elimin" protectia controalelor de la Panelul Principal;
            protectiePanel(false);
            //Initiez label Operatiunii cu "MODIFIC";
            lblOperatiune.Text = "MODIFIC";
            //Pun focus pe primul camp de completare;
            txtNumePacient.Focus();
        }
        private void A8(){
            OleDbConnection con = new OleDbConnection();
            OleDbCommand cmd = new OleDbCommand();
            con.ConnectionString =
internariTableAdapter.Connection.ConnectionString;
            cmd.Connection = con;//88->92 conectarea la baza de date + definirea
variabilelor in uz;
            cmd.CommandText = "DELETE FROM Internari WHERE IdInternare = " +
txtIdInternare.Text;//Comanda de tip SQL pentru stergerea Internarii;
            MessageBox.Show(cmd.CommandText);
            con.Open();
            cmd.ExecuteNonQuery();//Executia comenzii de stergere
            con.Close();
            refresh_grid(internariBindingSource.Position);//Am dat refresh la
grid, pentru a vizualiza in timp real stergerea internarii;
        }
        private void configureazaButoane(bool v){
            btnRenuntare.Visible = !v;
            btnConfirmare.Visible = !v;
            btnAdaugare.Visible = v;
            btnModificare.Enabled = v;
            btnStergere.Enabled = v;
        }
        private void legareControale(bool v){
            if (v){
                txtIdInternare.DataBindings.Add("Text", internariBindingSource,
"IdInternare");
                txtNumePacient.DataBindings.Add("Text", internariBindingSource,
"NumePacient");
                txtEpicriza.DataBindings.Add("Text", internariBindingSource,
"Epicriza");
                datainternare.DataBindings.Add("Value", internariBindingSource,
"DataInternarii");
```

```
                dataexternare.DataBindings.Add("Value", internariBindingSource,
"DataExternarii");
                comboSectie.DataBindings.Add("Text", internariBindingSource,
"Sectie");
                pozacardsanatate.DataBindings.Add("ImageLocation",
internariBindingSource, "PozaCard");
            }else{
                txtIdInternare.DataBindings.Clear();
                txtNumePacient.DataBindings.Clear();
                datainternare.DataBindings.Clear();
                comboSectie.DataBindings.Clear();
                dataexternare.DataBindings.Clear();
                pozacardsanatate.DataBindings.Clear();
                txtEpicriza.DataBindings.Clear();
            }
        }
        private void protectiePanel(bool v){
            txtNumePacient.ReadOnly = v;
            txtEpicriza.ReadOnly = v;
            comboSectie.Enabled = !v;
            datainternare.Enabled = !v;
            dataexternare.Enabled = !v;
            txtpozacard.ReadOnly = v;
        }
        private void golireCampuri(){
            txtNumePacient.Text = "";
            txtIdInternare.Text = "";
            txtEpicriza.Text = "";
            txtpozacard.Text = "";
            pozacardsanatate.ImageLocation = "";
        }
        private bool validareaCampurilor(){
            if(txtNumePacient.Text == ""){
                MessageBox.Show("Completați numele și prenumele pacientului");
                txtNumePacient.Focus();
                return false;
            }
            if(txtEpicriza.Text == ""){
                MessageBox.Show("Completați Epicriza");
                txtEpicriza.Focus();
                return false;
            }
            if(comboSectie.SelectedIndex == -1){
                MessageBox.Show("Completați secția unde pacientul se
internează!");
                comboSectie.Focus();
                return false;
            }
            return true;
        }
        private void adauga_internare(){
            string ListaAccess;//definesc o lista cu coloanele din Access
            string ListaText;//definesc o lista cu campurile text
            OleDbConnection con = new OleDbConnection();
            OleDbCommand cmd = new OleDbCommand();
            con.ConnectionString =
internariTableAdapter.Connection.ConnectionString;
            cmd.Connection = con;//realizez conectiunea la baza de date
            ListaAccess = "NumePacient, Sectie, DataInternarii, DataExternarii,
Epicriza, PozaCard";
            ListaText = "'"  + txtNumePacient.Text + "'" +
                        ",'" + comboSectie.SelectedValue + "'" +
                        ",'" + datainternare.Value + "'" +
```

```
                        ",'" + dataexternare.Value + "'" +
                        ",'" + txtEpicriza.Text + "'" +
                        ",'" + txtpozacard.Text + "'";
            cmd.CommandText = "INSERT INTO Internari(" + ListaAccess + ") " +
"SELECT " + ListaText;//comanda de tip SQL pentru Adaugarea in ACCESS
            MessageBox.Show(cmd.CommandText);
            con.Open();
            cmd.ExecuteNonQuery();//Executarea comenzii
            con.Close();
            ocupare_pat();//functia pentru ocuparea patului
        }
        private void ocupare_pat(){//REalizez o functie ce scade cu "-1" din
numarul Actual de paturi disponibile sectiei selectate
            OleDbConnection con = new OleDbConnection();
            OleDbCommand cmd = new OleDbCommand();
            con.ConnectionString =
internariTableAdapter.Connection.ConnectionString;
            cmd.Connection= con;//am realizat conexiunea la baza de date
            cmd.CommandText = "UPDATE Sectii SET NumarActualPaturi =
NumarActualPaturi -1 WHERE DSectie ='" + comboSectie.Text +"'";//comanda SQL
pentru a decrementa numarul Actual de paturi pentru Sectia respectiva
            con.Open();
            cmd.ExecuteNonQuery();//executia
            con.Close();

        }
        private void refresh_grid(int p){
            internariTableAdapter.Fill(bazadedate.Internari);
            internariBindingSource.Position = p;
        }
        private void modific_internare(){
            string lista;//initiez o lista, ce coreleaza campurile text cu
coloanele din Access
            OleDbConnection con = new OleDbConnection();
            OleDbCommand cmd = new OleDbCommand();
            con.ConnectionString =
internariTableAdapter.Connection.ConnectionString;
            cmd.Connection = con;//realizez conectiunea
            lista = "NumePacient ='" + txtNumePacient + "'," +
                    "Sectie ='" + comboSectie.SelectedValue + "'," +
                    "DataInternarii ='" + datainternare.Value + "'," +
                    "DataExternarii ='" + dataexternare.Value + "'," +
                    "Epicriza ='" + txtEpicriza.Text + "'," +
                    "PozaCard ='" + txtpozacard.Text + "'";
            cmd.CommandText = "UPDATE Internari SET " + lista + " WHERE
IdInternare=" + txtIdInternare.Text;//Comanda de tip SQL ce realizeaza
modificarea in Access;
            MessageBox.Show(cmd.CommandText);
            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();

        }
        private void btnAdaugare_Click(object sender, EventArgs e){
            A2();
        }

        private void btnModificare_Click(object sender, EventArgs e){
            A7();
        }
        private void btnStergere_Click(object sender, EventArgs e){
            const string Mesaj = "Confirmati stergerea internarii?";
            const string Titlu = "Stergere internare";
```

```
            var rez = MessageBox.Show(Mesaj, Titlu, MessageBoxButtons.YesNo,
MessageBoxIcon.Warning);
            if (rez == DialogResult.No) return;
            A8();
        }
        private void btnConfirmare_Click(object sender, EventArgs e){
            A4();
        }
        private void btnRenuntare_Click(object sender, EventArgs e){
            A3();
        }
        private void txtpozacard_MouseDoubleClick(object sender, EventArgs e){
            A6();
        }
        private void primiriUrgențeToolStripMenuItem_Click(object sender,
EventArgs e){
            Pacienti pacienti = new Pacienti();
            pacienti.Show();
            this.Hide();
        }
        private void rapoarteUPUToolStripMenuItem_Click(object sender, EventArgs
e){
            RapoarteUPU rupu = new RapoarteUPU();
            rupu.Show();
            this.Hide();
        }
        private void rapoarteInternăriToolStripMenuItem_Click(object sender,
EventArgs e){
            RapoarteInternari rint = new RapoarteInternari();
            rint.Show();
            this.Hide();
        }
    }
}
```

Form Pacienti/Primiri Urgente

```
using System;
using System.Data.OleDb;
using System.Windows.Forms;

namespace Proiect{
    public partial class Pacienti : Form{
        public Pacienti(){
            InitializeComponent();
        }
        private void Pacienti_Load(object sender, EventArgs e){
            this.codUrgentaTableAdapter.Fill(this.bazadedate1.CodUrgenta);
            A1();
        }
        private void A1(){
            this.pacientiTableAdapter.Fill(this.bazadedate.Pacienti);
            //Initiez protectia la nivelul gridului:
            dataGridView1.AllowUserToAddRows = false;
            dataGridView1.AllowUserToDeleteRows = false;
            dataGridView1.ReadOnly = true;
            A3();
        }
        private void A2(){
```

```
            //Configurez butoanele
            configureazaButoane(false);
            //Dezleg controalele
            legareControale(false);
            //"Ridic" protectia controalelor de la nivelul Panelului
            protectiePanel(false);
            //Modifc label de Operatiune in Adaugare lblOp
            lblOperatiune.Text = "ADAUG";
            //Pun "focus" pe primul camp de commpletare
            txtNume.Focus();
            // Golesc campurile
            golireCampuri();
        }
        private void A3(){
            //Initializez label pentru Operatiune drept null;
            lblOperatiune.Text = "";
            //Çonfigurez butoanele
            configureazaButoane(true);
            //Pun Protectie pe componentele din Panel
            protectiePanel(true);
            //Leg controalele
            legareControale(true);
        }
        private void A4(){
            if (lblOperatiune.Text == "ADAUG"){
                if (!validareCampuri()) return;//conditie ca campurile sa fie
completetate inainte de adaugare
                adauga_pacient();//chem functia de adaugare
                golireCampuri();//Golesc campurile, preg pentru urm adaugare
                txtNume.Focus();//Pun focus pe primul camp de completare

refresh_grid(pacientiBindingSource.Position);//Reactualizez/updatez grid-ul,
pentru a vizualiza in timp real Adaugarea;
            }else if (lblOperatiune.Text == "MODIFIC"){
                if (!validareCampuri()) return;//conditie ca campurile sa fie
completetate inainte de modificare
                modific_pacient();//chem functia de modificare

refresh_grid(pacientiBindingSource.Position);//Reactualizez/updatez grid-ul,
pentru a vizualiza in timp real Modificarea;
                A3();
            }
            else
                MessageBox.Show("Operatie actualizare neefectuata");
        }
        private void A5(){
            //Configurez butoanele
            configureazaButoane(false);
            //Dezleg controalele din Panel
            legareControale(false);
            //Ridic protectia pusa pe controalele din Panel
            protectiePanel(false);
            //Modifc label-ul pentru Operatiune cu 'Modific'
            lblOperatiune.Text = "MODIFIC";
            //Pun focus pe primul camp de completare
            txtNume.Focus();
        }
        private void A6(){
            OleDbConnection con = new OleDbConnection();
            OleDbCommand cmd = new OleDbCommand();
            //OleDbDataReader rdr;
            con.ConnectionString =
pacientiTableAdapter.Connection.ConnectionString;
```

```
            cmd.Connection = con;//82->86 am realizat conexiunea la baza de date
cu variabilele in uz
            cmd.CommandText = "DELETE FROM Pacienti WHERE IdPacient = " +
txtIdPacient.Text;//comanda de tip SQL ce realizeaza stergerea din baza de date
ACCESS
            MessageBox.Show(cmd.CommandText);
            con.Open();
            cmd.ExecuteNonQuery();//execut comanda SQL
            con.Close();
            refresh_grid(pacientiBindingSource.Position);//Reactualizez/updatez
grid-ul, pentru a vizualiza in timp real Stergerea;
        }
        private void configureazaButoane(bool v){
            btnRenuntare.Visible = !v;
            btnConfirmare.Visible = !v;
            btnAdaugare.Enabled = v;
            btnModificare.Enabled = v;
            btnStergere.Enabled = v;
        }
        private void legareControale(bool v){
            if (v){
                txtIdPacient.DataBindings.Add("Text", pacientiBindingSource,
"IdPacient");
                txtNume.DataBindings.Add("Text", pacientiBindingSource,
"NumePacient");
                txtTelefon.DataBindings.Add("Text", pacientiBindingSource,
"NrTelefon");
                txtCNP.DataBindings.Add("Text", pacientiBindingSource, "CNP");
                txtDiagnostic.DataBindings.Add("Text", pacientiBindingSource,
"Diagnostic");
                comboCod.DataBindings.Add("Text", pacientiBindingSource,
"CodUrgenta");
                comboDecizie.DataBindings.Add("Text", pacientiBindingSource,
"Decizie");
            }else{
                txtIdPacient.DataBindings.Clear();
                txtNume.DataBindings.Clear();
                txtCNP.DataBindings.Clear();
                txtTelefon.DataBindings.Clear();
                txtDiagnostic.DataBindings.Clear();
                comboCod.DataBindings.Clear();
                comboDecizie.DataBindings.Clear();
            }
        }
        private void protectiePanel(bool v){
            txtNume.ReadOnly = v;
            txtCNP.ReadOnly = v;
            txtDiagnostic.ReadOnly = v;
            txtTelefon.ReadOnly = v;
            comboCod.Enabled = !v;
            comboDecizie.Enabled = !v;
        }
        private void golireCampuri(){
            txtIdPacient.Text = "";
            txtNume.Text = "";
            txtCNP.Text = "";
            txtTelefon.Text = "";
            txtDiagnostic.Text = "";
            comboDecizie.SelectedIndex = 0;
            comboCod.SelectedIndex = 0;
        }
        private void refresh_grid(int p){
            pacientiTableAdapter.Fill(bazadedate.Pacienti);
```

```
            pacientiBindingSource.Position = p;
        }
        private bool validareCampuri(){
            if(txtNume.Text == ""){
                MessageBox.Show("Completati numele și prenumele pacientului.");
                txtNume.Focus();
                return false;
            }
            if (txtTelefon.Text == ""){
                MessageBox.Show("Completati numărul de telefon al pacientului.");
                txtTelefon.Focus();
                return false;
            }
            if (txtDiagnostic.Text == ""){
                MessageBox.Show("Completati diagnostic de urgență al
pacientului.");
                txtDiagnostic.Focus();
                return false;
            }
            if(txtCNP.Text == ""){
                MessageBox.Show("Completati Codul Numeric Personal al
pacientului.");
                txtCNP.Focus();
                return false;
            }
            if (comboCod.SelectedIndex == -1) {
                MessageBox.Show("Completati codul de urgență al pacientului.");
                comboCod.Focus();
                return false;
            }
            if (comboDecizie.SelectedIndex == -1){
                MessageBox.Show("Completati decizia actuală a pacientului.");
                comboDecizie.Focus();
                return false;
            }
            return true;
        }
        private void adauga_pacient(){
            string ListaAccess;//Initiez o lista cu valorile coloanelor din
access
            string ListaText;//initiez o lista cu controalele/campurile de
completat
            OleDbConnection con = new OleDbConnection();
            OleDbCommand cmd = new OleDbCommand();
            con.ConnectionString =
pacientiTableAdapter.Connection.ConnectionString;
            cmd.Connection = con;//initiez conexiunea la baza de date
            ListaAccess = "NumePacient, CNP, NrTelefon, CodUrgenta, Diagnostic,
Decizie";
            ListaText = "'" + txtNume.Text + "'" +
                        ",'" + txtCNP.Text + "'" +
                        ",'" + txtTelefon.Text + "'" +
                        ",'" + comboCod.SelectedValue + "'"+
                        ",'" + txtDiagnostic.Text + "'" +
                        ",'" + comboDecizie.SelectedItem + "'";
            cmd.CommandText = "INSERT INTO Pacienti(" + ListaAccess + ") " + "
SELECT " + ListaText ;//comanda de tip SQL pentru Adaugarea pacientului in baza
de date;
            MessageBox.Show(cmd.CommandText);
            con.Open();
            cmd.ExecuteNonQuery();//execut comanda SQL
            con.Close();
        }
```

```
        private void modific_pacient(){
            string Lista;
            OleDbConnection con = new OleDbConnection();
            OleDbCommand cmd = new OleDbCommand();
            con.ConnectionString =
pacientiTableAdapter.Connection.ConnectionString;
            cmd.Connection = con;
            Lista = "NumePacient = '" + txtNume.Text + "'," +
                    "NrTelefon = '" + txtTelefon.Text + "'," +
                    "Diagnostic = '" + txtDiagnostic.Text + "'," +
                    "CNP = '" + txtCNP.Text + "'," +
                    "CodUrgenta = '" + comboCod.SelectedValue + "'," +
                    "Decizie = '" + comboDecizie.SelectedItem + "'";
            cmd.CommandText = "UPDATE Pacienti SET " + Lista + " WHERE
IdPacient=" + txtIdPacient.Text;//comanda de tip SQL pentru Modificarea
pacientului in baza de date;
            MessageBox.Show(cmd.CommandText);
            con.Open();
            cmd.ExecuteNonQuery();//execut comanda SQL
            con.Close();
        }
        private void internăriToolStripMenuItem_Click(object sender, EventArgs
e){
            Internari intern = new Internari();
            intern.Show();
            this.Hide();
        }
        private void rapoarteUPUToolStripMenuItem_Click(object sender, EventArgs
e){
            RapoarteUPU rupu = new RapoarteUPU();
            rupu.Show();
            this.Hide();
        }
        private void rapoarteInternăriToolStripMenuItem_Click(object sender,
EventArgs e){
            RapoarteInternari rint = new RapoarteInternari();
            rint.Show();
            this.Hide();
        }
        private void btnConfirmare_Click(object sender, EventArgs e){
            A4();
        }
        private void btnRenuntare_Click(object sender, EventArgs e){
            A3();
        }
        private void btnAdaugare_Click(object sender, EventArgs e){
            A2();
        }
        private void btnModificare_Click(object sender, EventArgs e){
            A5();
        }
        private void btnStergere_Click(object sender, EventArgs e){
            const string msj = "Confirmati ștergerea pacientului?";
            const string Titlu = "Ștergere pacient UPU!";
            var rezultat = MessageBox.Show(msj, Titlu,
            MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
            if (rezultat == DialogResult.No) return;
            A6();
        }
    }
}
```

Form Raporte Internari:

```
using System;
using System.Data;
using System.Windows.Forms;
using System.Windows.Forms.DataVisualization.Charting;
using System.Data.OleDb;

namespace Proiect{
    public partial class RapoarteInternari : Form{
        private OleDbConnection con = new
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\\Users\\olaru\\Desktop\\ProiectMPP2\\Resurse\\HOS2.mdb");
        private OleDbCommand cmd = new OleDbCommand();
        private OleDbDataReader rdr;
        private int variabilaLinii;
        public RapoarteInternari(){
            InitializeComponent();
        }
        private void filtru(){
            try{internariBindingSource1.Filter = "Sectie = '" + comboBox1.Text +
"'";}
            catch{ }
        }
        private void RapoarteInternari_Load(object sender, EventArgs e){
            this.queryIntPatTableAdapter.Fill(this.bazadedate.QueryIntPat);
            this.internariTableAdapter.Fill(this.bazadedate.Internari);
            this.sectiiTableAdapter.Fill(this.bazadedate.Sectii);
            filtru();
            chart();
        }
        private void comboBox1_SelectedIndexChanged(object sender, EventArgs e){
            filtru();
        }

        private void chart(){
            variabilaLinii = bazadedate.QueryIntPat.Count;
            int[] yval = new int[variabilaLinii];
            string[] xval = new string[variabilaLinii];
            int i = 0;
            foreach (DataRowView r in queryIntPatBindingSource){
                xval[i] = r["Sectie"].ToString();
                 yval[i] = Convert.ToInt32(r["NumarActualPaturi"].ToString());
                 i++;
            }
            chartPat.Series[0].Points.DataBindXY(xval, yval);
            chartPat.Series[0].Name = "Situație Paturi Sectii";
            chartPat.Series[0].ChartType = SeriesChartType.Column;
            chartPat.Invalidate();
        }

        private void btnRaport_Click(object sender, EventArgs e){
            RaportrdlcInternari intrdlc = new RaportrdlcInternari();
            intrdlc.Show();
        }

        private void btnRaportPaturi_Click(object sender, EventArgs e){
            RaportrdlcInternariPaturi intpatrdlc = new
RaportrdlcInternariPaturi();
            intpatrdlc.Show();
        }
```

```
        private void btnFiltruData_Click(object sender, EventArgs e){
            internariDataTableAdapter.Fill(bazadedate.InternariData,
dateTimePicker1.Value, dateTimePicker2.Value);
        }
        private void primiriUrgențeToolStripMenuItem_Click_1(object sender,
EventArgs e){
            Pacienti UPU = new Pacienti();
            UPU.Show();
            this.Hide();
        }
        private void internăriToolStripMenuItem_Click(object sender, EventArgs
e){
            Internari intern = new Internari();
            intern.Show();
            this.Hide();
        }
        private void rapoarteUPUToolStripMenuItem_Click(object sender, EventArgs
e){
            RapoarteUPU rupu = new RapoarteUPU();
            rupu.Show();
            this.Hide();
        }
        private void rapoarteInternăriToolStripMenuItem_Click(object sender,
EventArgs e){
            RapoarteInternari rrupu = new RapoarteInternari();
            rrupu.Show();
            this.Hide();
        }
    }
}
```

Form Rapoarte UPU:

```
using System;
using System.Windows.Forms.DataVisualization.Charting;
using System.Data;
using System.Windows.Forms;

namespace Proiect{
    public partial class RapoarteUPU : Form{
        private int variabilaLinii;
        public RapoarteUPU(){
            InitializeComponent();
        }
        private void RapoarteUPU_Load(object sender, EventArgs e){
            this.queryUPUTableAdapter.Fill(this.bazadedate.QueryUPU);
            this.pacientiTableAdapter.Fill(this.bazadedate.Pacienti);
            this.codUrgentaTableAdapter.Fill(this.bazadedate.CodUrgenta);
            filtruCombo();
            chart();
        }
        private void primiriUrgențeToolStripMenuItem_Click(object sender,
EventArgs e){
            Pacienti pac = new Pacienti();
            pac.Show();
            this.Hide();
        }
```

```
        private void internăriToolStripMenuItem_Click(object sender, EventArgs
e){
            Internari intern = new Internari();
            intern.Show();
            this.Hide();
        }
        private void rapoarteInternăriToolStripMenuItem_Click(object sender,
EventArgs e){
            RapoarteInternari rint = new RapoarteInternari();
            rint.Show();
            this.Hide();
        }
        private void filtruCombo(){
            try { pacientiBindingSource.Filter = "CodUrgenta= '" + comboCod.Text
+ "'"; }
            catch { };
        }

        private void chart(){
            variabilaLinii = bazadedate.QueryUPU.Count;
            int[] yval = new int[variabilaLinii];
            string [] xval = new string[variabilaLinii];
            int i = 0;
            foreach (DataRowView r in queryUPUBindingSource){
                xval[i] = r["CodUrgenta"].ToString();
                yval[i] = Convert.ToInt32(r["TotalIntrari"].ToString());
                i++;
            }
            chartCod.Series[0].Points.DataBindXY(xval, yval);
            chartCod.Series[0].Name = "Situație Coduri UPU";
            chartCod.Series[0].ChartType = SeriesChartType.Column;
            chartCod.Invalidate();
        }
        private void btnRaport_Click(object sender, EventArgs e){
            RaportrdlcUPU upurdlc = new RaportrdlcUPU();
            upurdlc.Show();
        }

        private void comboCod_SelectedIndexChanged(object sender, EventArgs e)
        {
            filtruCombo();
        }
    }
}
```

Form-uri pentru RDLC:

Raport Internari:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace Proiect{
    public partial class RaportrdlcInternari : Form{
        public RaportrdlcInternari(){
            InitializeComponent();
        }
        private void RaportrdlcInternari_Load(object sender, EventArgs e){
            // TODO: This line of code loads data into the 'bazadedate.Internari'
table. You can move, or remove it, as needed.
            this.internariTableAdapter.Fill(this.bazadedate.Internari);
            this.reportViewer1.RefreshReport();
            this.reportViewer1.RefreshReport();
            this.reportViewer1.RefreshReport();
        }
    }
}
```

Raport paturi:

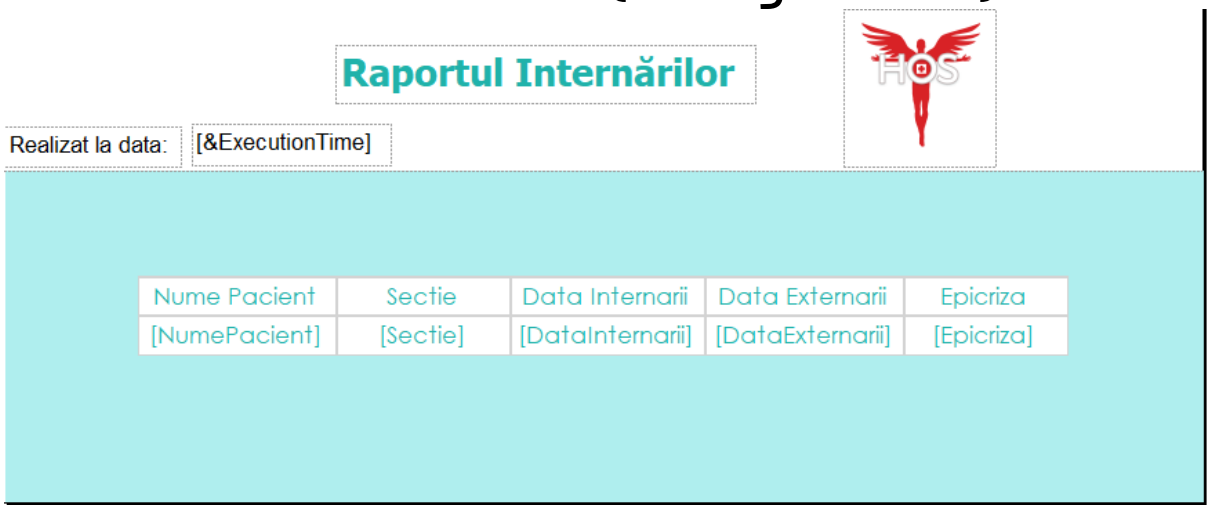
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Proiect{
    public partial class RaportrdlcInternari : Form{
        public RaportrdlcInternari(){
            InitializeComponent();
        }
        private void RaportrdlcInternari_Load(object sender, EventArgs e){
            // TODO: This line of code loads data into the 'bazadedate.Internari'
table. You can move, or remove it, as needed.
            this.internariTableAdapter.Fill(this.bazadedate.Internari);
            this.reportViewer1.RefreshReport();
            this.reportViewer1.RefreshReport();
            this.reportViewer1.RefreshReport();
        }
    }
}
```

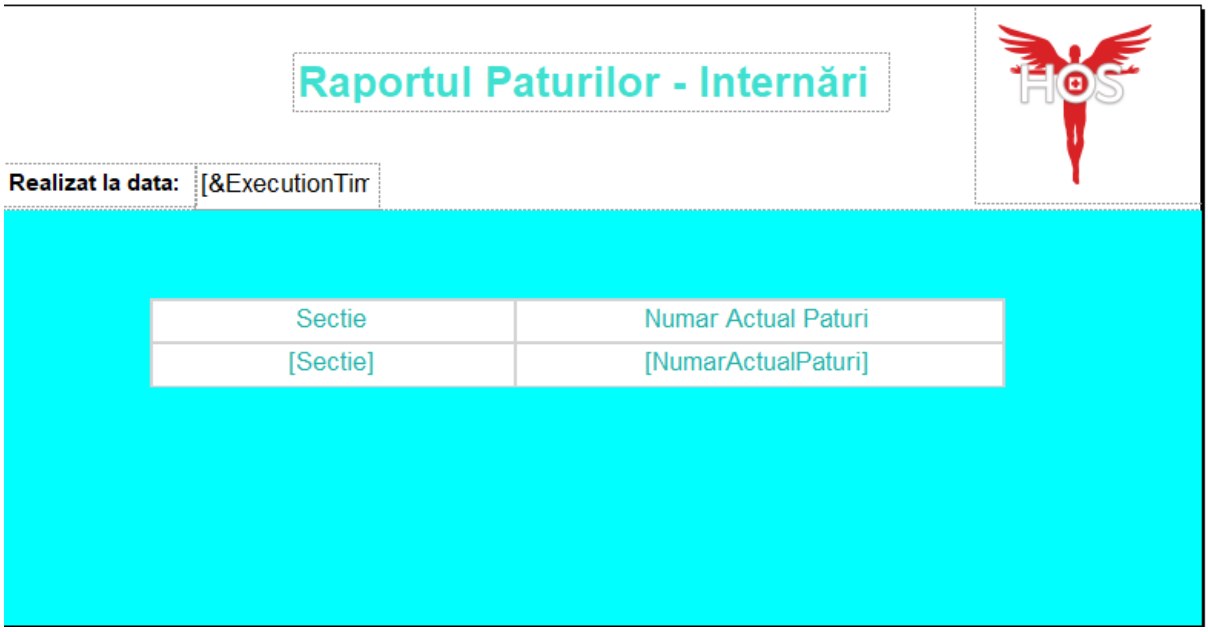
Raport UPU:

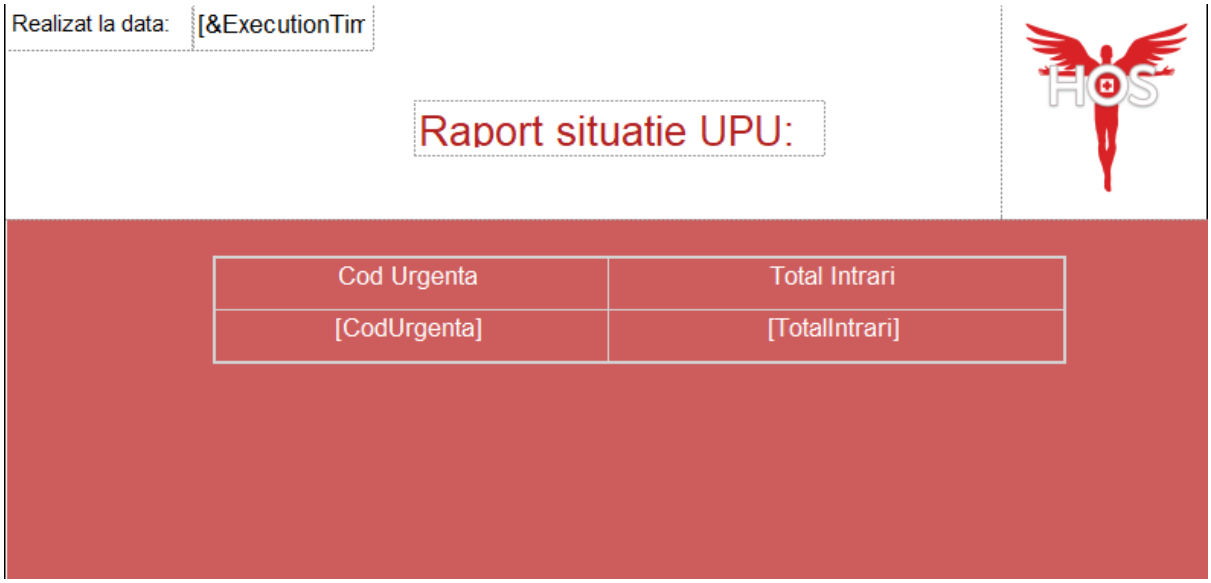
```
using System;

using System.Data;
using System.Windows.Forms;

namespace Proiect{
    public partial class RaportrdlcUPU : Form{
        public RaportrdlcUPU(){
            InitializeComponent();
        }
        private void RaportParasireUPU_Load(object sender, EventArgs e){
            // TODO: This line of code loads data into the 'bazadedate.QueryUPU'
table. You can move, or remove it, as needed.
            this.queryUPUTableAdapter.Fill(this.bazadedate.QueryUPU);
```

```
            this.reportViewer1.RefreshReport();
        }
    }
}
```

RAPOARTE RDLC(Design view)






Realizat la data:
[&ExecutionTim
Raport situatie UPU:
Cod Urgenta
Total Intrari
[CodUrgenta]
[TotalIntrari]