

**Colegiul Național „Spiru Haret”**

**LUCRARE DE ATESTAT  
LA INFORMATICĂ**

Profesor coordonator,  
**Flavius Boian**

Elev,  
**Sebastian Popa**

Târgu Jiu  
2022

**Colegiul Național „Spiru Haret”**

**LUCRARE DE ATESTAT  
LA INFORMATICĂ**

**SISTEM DE EVIDENȚĂ ȘCOLARĂ**

Profesor coordonator,  
**Flavius Boian**

Elev,  
**Sebastian Popa**

Târgu Jiu  
2022

# Cuprins

- 1. Motivație**
- 2. Tehnologii**
- 3. Funcționalitate**
- 4. Atașamente**
- 5. Cod sursă**
- 6. Bibliografie**

## **1. Motivație**

Scopul acestui proiect este de a ușura munca unui secretar în ceea ce privește menținerea evidenței elevilor, profesorilor și taxelor dintr-o școală. Proiectul permite stocarea în mod eficient și intuitiv a datelor referitoare la elevii și profesorii unei școli (nume, date de contact, adresa, varsta), dar și la taxele plătite de elevi (descriere, suma plătită, suma totală).

## 2. Tehnologii

Pe întreg parcursul dezvoltării aplicației s-a încercat folosirea celor mai adecvate tehnologii și limbaje de programare pentru îndeplinirea obiectivelor. Astfel, pentru a păstra lizibilitatea codului, portabilitatea aplicației, dar și eficiența în timp și spațiu, am folosit următoarele tehnologii și aplicații:

- .NET framework
- .Net core
- Windows forms
- C# programming language
- Visual Studio
- Visual Studio Code
- SQL Server Management Studio
- SQL
- GitHub
- Git

### **3. Funcționalitate**

Pentru a facilita gestionarea datelor referitoare la elevi, profesori și taxe în mod eficient, aplicația permite adăugarea, ștergerea, modificarea și căutarea unui profesor, unui elev sau a unei taxe.

În momentul adăugării unei intrări noi, datele sunt verificate pentru corectitudinea lor, iar în cazul unei erori utilizatorul este atenționat. Exemple: nume ce conține cifre, dată de naștere din viitor, număr de telefon ce conține litere.

Căutarea unui elev, unui profesor sau a unei taxe se poate realiza după unul dintre câmpuri, de exemplu după numele persoanei, după email, după adresă, după materia predată sau după numărul de telefon.

Aplicația poate crea un tip de taxă și înregistra o plată parțială sau integrală a unei taxe efectuată de către un elev. În cazul în care o taxă a fost plătită integral, aceasta este eliminată din taxele active și adăugată într-un registru de taxe plătite.

## 4. Atașamente

În cele ce urmează voi prezenta fragmente de cod semnificative pentru funcționalitățile aplicației.

Pentru inserarea, actualizarea și ștergerea datelor în baza de date inițializez o conexiune cu baza de date și creez o comandă SQL pe care ulterior o voi popula cu datele introduse de utilizator.

Form titled "Date Personale" with the following fields:

- Clasa:
- Sex: ☐ Masculin ☐ Feminin
- Nume:  Preume:
- Email:  Nr. Tlf.:
- Adresa:
- Data Nasterii:  Data inregistrarii:
- Judet:
- Localitate:

Procesare sidebar:

- Golire
- Adaugare
- Actualizare
- Stergere
- Date

Form titled "Date Personale" with the following fields:

- Materie:
- Sex: ☐ Masculin ☐ Feminin
- Nume:  Preume:
- Email:  Nr. Tlf.:
- Adresa:
- Data Nasterii:  Experienta:
- Judet:
- Localitate:

Procesare sidebar:

- Golire
- Adaugare
- Actualizare
- Stergere
- Date

```
if (con.State != ConnectionState.Open)
    con.Open();
string sqr = "insert into elevi values(" +
    "@clasa," + "@sex," + "@nume," + "@preume," + "@email," +
    "@nrTlf," + "@adresa," + "@datanast," + "@datainreg," +
    "@judet," + "@idjud," + "@oras," + "@idoras" + ")";
SqlCommand cmd;
try
{cmd= new SqlCommand(sqr, con);}
catch (Exception es)
{
    MessageBox.Show(es.ToString());return;
}
```

Pentru gestionarea taxelor este posibilă adăugarea unei taxe atribuite unui elev, dar și înregistrarea unei plăți parțiale sau integrale a unei taxe.

←

Adaugare taxa

Platire taxa

ID	Denumire	Descriere
3	Taxa Creta	Taxa lunara pentru creta
7	Taxa de inscriere	Inscriere la liceu
8	Fondul clasei	Bani pentru fondul clasei

ID TAXA SELECTATA

15

Platire

Anulare Plata

Valoare taxa

Vezi toti elevii

Adauga Taxa

	Taxa	Valoare	Platit
►	Taxa Creta	15	0

id

Nume

Prenume

Clasa

Nr. de tf.

►	32	Popa	Sebastian	12	0767676767
	33	Vasiluta	Alexandru	10	0773747169

Cautare

ID ELEV SELECTAT

32


```
private void BPlata_Click(object sender, EventArgs e)
{
    if(String.IsNullOrEmpty(LTaxa.Text))
    {
        MessageBox.Show("Nu ati selectat o taxa", "Eroare", MessageBoxButtons.OK);
        return;
    }
    int ind = Int16.Parse(LTaxa.Text);
    int valoare = 0, platit = 0;
    GetVaP(ref valoare, ref platit);
    if (!ValidNr(TBPlata.Text))
    {
        MessageBox.Show("Valoare invalida", "Eroare", MessageBoxButtons.OK);
        return;
    }
    int nval = Int32.Parse(TBPlata.Text);
    if (nval + platit > valoare)
    {
        MessageBox.Show("Suma este prea mare", "Eroare", MessageBoxButtons.OK);
        return;
    }
    nval += platit;
    AlterTaxa(ind, nval);

    if(nval==valoare)
    {
        try
        {
            int id = Int32.Parse(LTaxa.Text);
            int idelev = Int32.Parse(LElev.Text);
            //valoare
            int idtaxa = GetTipTaxa();
            if (con.State != ConnectionState.Open)
                con.Open();
            SqlCommand cmd = new SqlCommand("insert into istorictaxe values (@id,@idtaxa,@idelev,@valoare)", con);
            cmd.Parameters.AddWithValue("@id", id);
            cmd.Parameters.AddWithValue("@idtaxa", idtaxa);
            cmd.Parameters.AddWithValue("@idelev", idelev);
            cmd.Parameters.AddWithValue("@valoare", valoare);
            cmd.ExecuteNonQuery();

            SqlCommand ncmd = new SqlCommand("delete from taxe where id =@id", con);
            LTaxa.Text = "";
            ncmd.Parameters.AddWithValue("@id", id);
            ncmd.ExecuteNonQuery();
            con.Close();
            MessageBox.Show("Taxa a fost platita in totalitate");
            PuneTaxe();
        }
    }
}
```



Aplicația permite, de asemenea, vizualizarea datelor sub formă de tabele.



Adaugare Taxe

ID taxa selectata

Denumire

Descriere

Adaugare

Stergere

Taxe

ID	Denumire	Descriere
8	Fondul clasei	Bani pentru fondul clasei
3	Taxa Creta	Taxa lunara pentru creta
7	Taxa de inscriere	Inscriere la liceu

Istoric Taxe

Nume	Prenume	Email	Taxa	Valoare
Popa	Sebastian	sebastian@mail...	Taxa Creta	50

```
try
{
    if (con.State != ConnectionState.Open)
        con.Open();
    LV.BackgroundImage = null;
    LV.Columns.Clear();
    LV.Items.Clear();
    LV.Columns.Add("ID", 40, HorizontalAlignment.Center);
    LV.Columns.Add("Denumire", 220, HorizontalAlignment.Center);
    LV.Columns.Add("Descriere", 220, HorizontalAlignment.Center);
    LV.View = View.Details;
    SqlCommand cmd = new SqlCommand("select *from tiptaxe order by denumire asc", con);
    SqlDataAdapter adp = new SqlDataAdapter(cmd);
    DataSet ds = new DataSet();
    adp.Fill(ds, "test");
    DataTable dt = ds.Tables["test"];
    for (int i = 0; i < dt.Rows.Count; i++)
    {
        LV.Items.Add(dt.Rows[i].ItemArray[0].ToString());
        LV.Items[i].SubItems.Add(dt.Rows[i].ItemArray[1].ToString());
        LV.Items[i].SubItems.Add(dt.Rows[i].ItemArray[2].ToString());
    }
    con.Close();
}
catch (Exception ee)
{
    MessageBox.Show(ee.ToString());
}
```

## 5. Cod sursă

În cele ce urmează, voi insera fragmente de cod vitale pentru funcționarea aplicației.

```
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 SistemDeEvidenta
{
    public partial class FereastrPrincipala : Form
    {
        public FereastrPrincipala()
        {
            InitializeComponent();
        }

        private void BElevi_Click(object sender, EventArgs e)
        {
            this.Hide();
            Elevi form2 = new Elevi();
            form2.Show();
        }

        private void BProfesori_Click(object sender, EventArgs e)
        {
            this.Hide();
            Profesori form2 = new Profesori();
            form2.Show();
        }

        private void BTaxa_Click(object sender, EventArgs e)
        {
            this.Hide();
            Taxe form2 = new Taxe();
            form2.Show();
        }

        private void BRapoarte_Click(object sender, EventArgs e)
        {
            this.Hide();
            Rapoarte form2 = new Rapoarte();
            form2.Show();
        }

        private void BAdmin_Click(object sender, EventArgs e)
        {
            this.Hide();
            Admin form2 = new Admin();
            form2.Show();
        }
    }
}
```

```

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;
using System.Data.SqlClient;
using System.Data.SqlTypes;
using System.Diagnostics;

namespace SistemDeEvidenta
{
    public partial class Elevi : Form
    {
        public static string CONSTRING = ClasaConString.ConString();
        //aduag asta aici ca sa fie
        readonly SqlConnection con = new SqlConnection(CONSTRING);
        public Elevi()
        {
            InitializeComponent();
        }
        private void BBack_Click(object sender, EventArgs e)
        {
            this.Close();
            Application.OpenForms["FereastraPrincipala"].Show();
        }
        public void PuneJudete()
        {
            if (con.State != ConnectionState.Open)
                con.Open();
            SqlCommand cmd = new SqlCommand("select judet from judete order by judet
asc", con);
            SqlDataReader rdr = cmd.ExecuteReader();
            DataTable dt = new DataTable();
            dt.Columns.Add("judet", typeof(string));
            dt.Load(rdr);
            CBJudet.ValueMember = "judet";
            CBJudet.DataSource = dt;
            con.Close();
            CBJudet_SelectionChangeCommitted(this, EventArgs.Empty);
        }
        private void Elevi_Load(object sender, EventArgs e)
        {
            try
            {
                PuneJudete();
                CBJudet.SelectedIndex = -1;
                CBOras.SelectedIndex = -1;
            }
            catch (Exception ee)
            {
                MessageBox.Show(ee.ToString());
            }
        }
    }
}

```

```

private void BGolire_Click(object sender, EventArgs e)
{
    TBClasa.Text = "";
    RBMasc.Checked = RBFem.Checked = false;
    TBEmail.Text = TBTlf.Text = "";
    TBAdresa.Text = "";
    TBNuma.Text = TBPrename.Text = "";
    DatePickerNastere.Value = DateTime.Now;
    DateTimeInreg.Value = DateTime.Now;
    LElev.Text = "";
    try
    {
        PuneJudete();
    }
    catch (Exception ee)
    {
        MessageBox.Show(ee.ToString());
    }
}

public int GetIdOras()
{
    string oras = CBOoras.Text;
    if (con.State != ConnectionState.Open)
        con.Open();
    string sqr = $"select id from orase where oras='{oras}'";
    SqlCommand cmd = new SqlCommand(sqr, con);
    SqlDataReader rdr = cmd.ExecuteReader();
    DataTable dt = new DataTable();
    dt.Columns.Add("id", typeof(int));
    dt.Load(rdr);
    int id = Int32.Parse(dt.Rows[0]["id"].ToString());
    return id;
}

public int GetIdJudet()
{
    string judet = CBJudet.Text;
    if (con.State != ConnectionState.Open)
        con.Open();
    string sqr = $"select id from judete where judet='{judet}'";
    SqlCommand cmd = new SqlCommand(sqr, con);
    SqlDataReader rdr = cmd.ExecuteReader();
    DataTable dt = new DataTable();
    dt.Columns.Add("id", typeof(int));
    dt.Load(rdr);
    int id = Int32.Parse(dt.Rows[0]["id"].ToString());
    return id;
}

public bool VerifSex()
{
    if (!RBMasc.Checked && !RBFem.Checked)
        return false;
    return true;
}

public bool ValidareDate()
{
    if (!ClasaValidareDate.VerifClasa(TBClasa.Text))

```

```

    MessageBox.Show("Clasa invalida.", "Eroare", MessageBoxButtons.OK);
        return false;
    }
    if (!VerifSex())
    {
        MessageBox.Show("Nu ati introdus sexul.", "Eroare",
    MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifNume(TBNuma.Text))
    {
        MessageBox.Show("Nume invalid", "Eroare", MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifPrenume(TBPrenume.Text))
    {
        MessageBox.Show("Prenume invalid", "Eroare", MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifEmail(TBEmail.Text))
    {
        MessageBox.Show("Email invalid", "Eroare", MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifNrTlf(TBTlf.Text))
    {
        MessageBox.Show("Numar de telefon invalid", "Eroare",
    MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifAdresa(TBAdresa.Text))
    {
        MessageBox.Show("Adresa invalida", "Eroare", MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifData(DatePickerNastere))
    {
        MessageBox.Show("Data de nastere invalida", "Eroare",
    MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifData(DateTimeInreg))
    {
        MessageBox.Show("Data de Inregistrare invalida", "Eroare",
    MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifCB(CBJudet))
    {
        MessageBox.Show("Judet invalid", "Eroare", MessageBoxButtons.OK);
        return false;
    }
    if (!ClasaValidareDate.VerifCB(CBOras))
    {
        MessageBox.Show("Oras invalid", "Eroare", MessageBoxButtons.OK);
        return false;
    }
    return true;
}

```

```

private void BCautare_Click(object sender, EventArgs e)
{
    var mmap = new Dictionary<string, string>(){
        { "Nume", "nume" },
        { "Prenume", "prenume" },
        { "Clasa", "clasa" },
        { "Nr. Tlf.", "nrtlf" },
        { "Email", "email" },
        { "Judet", "judet" },
        { "Adresa", "adresa" },
        { "Sex", "sex" },
        { "Oras", "oras" }};
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        if (string.IsNullOrEmpty(CBCriteriu.Text) || CBCriteriu.SelectedIndex ==
-1)
        {
            MessageBox.Show("Nu ati ales criteriu de
cautare", "Eroare", MessageBoxButtons.OK);
            return;
        }
        if (string.IsNullOrEmpty(TBCriteriu.Text))
        {
            MessageBox.Show("Nu ati introdus date de cautare", "Eroare",
MessageBoxButtons.OK);
            return;
        }

        SqlDataAdapter adp = new SqlDataAdapter($"select id, nume as
'Nume', prenume as 'Prenume', clasa as 'Clasa', nrtlf as 'Nr. de tlf.', email as
'Email', judet as 'Judet', oras as 'Oras', adresa as 'Adresa' " +
        $"", sex as 'Sex', dnastere as 'Data de nastere', dinregistrare as 'Data
de inregistrare' from elevi where {mmap[CBCriteriu.Text]} like '%" + TBCriteriu.Text+"%' ",
con);

        System.Data.DataTable tabel = new System.Data.DataTable();
        adp.Fill(tabel);
        con.Close();
        DGVCriteriu.DataSource = tabel;
    }
    catch (Exception ee)
    {
        MessageBox.Show(ee.ToString());
    }
}

private void DGVCriteriu_CellDoubleClick(object sender, DataGridViewCellEventArgs
e)
{
    int ind = e.RowIndex;
    if (ind < 0) return;
    LElev.Text = DGVCriteriu.Rows[ind].Cells[0].Value.ToString();

    TBClasa.Text = DGVCriteriu.Rows[ind].Cells[3].Value.ToString();

    bool valsex =(bool) DGVCriteriu.Rows[ind].Cells[9].Value;
    if (valsex)

```

```

RBMasc.Checked = true;
    RBFem.Checked = false;
}
else
{
    RBMasc.Checked = false;
    RBFem.Checked = true;
}

TBNuma.Text = DGVCriteria.Rows[ind].Cells[1].Value.ToString();
TBPrename.Text = DGVCriteria.Rows[ind].Cells[2].Value.ToString();
TBEmail.Text = DGVCriteria.Rows[ind].Cells[5].Value.ToString();
TBTlf.Text = DGVCriteria.Rows[ind].Cells[4].Value.ToString();
TBAdresa.Text = DGVCriteria.Rows[ind].Cells[8].Value.ToString();
DatePickerNastere.Value = Convert.ToDateTime(
DGVCriteria.Rows[ind].Cells[10].Value);
DateTimeInreg.Value = Convert.ToDateTime(
DGVCriteria.Rows[ind].Cells[11].Value);
CBJudet.SelectedIndex =
CBJudet.FindStringExact(DGVCriteria.Rows[ind].Cells[6].Value.ToString());
CBJudet_SelectionChangeCommitted(this, EventArgs.Empty);
CBOras.SelectedIndex =
CBOras.FindStringExact(DGVCriteria.Rows[ind].Cells[7].Value.ToString());
}

private void Bstergere_Click(object sender, EventArgs e)
{
    string text = LElev.Text;
    if (string.IsNullOrEmpty(text))
    {
        MessageBox.Show("Nu ati selectat niciun elev.", "Eroare",
MessageBoxButtons.OK);
        return;
    }
    try
    {
        int id = Int16.Parse(LElev.Text);
        DialogResult dr = MessageBox.Show($"Stergeti elevul cu id-ul {id}?",
"Confirmare", MessageBoxButtons.YesNo);
        if (dr == DialogResult.Yes)
        {
            if (con.State != ConnectionState.Open)
                con.Open();
            SqlCommand cmd = new SqlCommand($"delete from elevi where id={id}",
con);

            cmd.ExecuteNonQuery();
            MessageBox.Show("Stergere reusita");
            BDate_Click(this, EventArgs.Empty);
            con.Close();
        }
    }
    catch (Exception es)
    {
        MessageBox.Show(es.ToString());
    }
}

```

```

private void BDate_Click(object sender, EventArgs e)
{
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        SqlDataAdapter adp = new SqlDataAdapter($"select id, nume as
'Nume',prenume as 'Prenume',clasa as 'Clasa',nr_tlf as 'Nr. de tlf.',email as
'Email',judet as 'Judet',oras as 'Oras',adresa as 'Adresa' " +
            $",sex as 'Sex',dnastere as 'Data de nastere',dinregistrare as 'Data
de inregistrare' from elevi", con);
        System.Data.DataTable tabel = new System.Data.DataTable();
        adp.Fill(tabel);
        con.Close();
        DGVCriteria.DataSource = tabel;
    }
    catch (Exception ee)
    {
        MessageBox.Show(ee.ToString());
    }
}

private void Elevi_FormClosing(object sender, FormClosingEventArgs e)
{
    if (new StackTrace().GetFrames().Any(x => x.GetMethod().Name == "Close"))
        return;
    else
        Application.OpenForms["FereastraPrincipala"].Close();
}
}

```



```

private void Taxe_Load(object sender, EventArgs e)
{
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        listView1.Columns.Add("ID", 30, HorizontalAlignment.Center);
        listView1.Columns.Add("Denumire", 180, HorizontalAlignment.Center);
        listView1.Columns.Add("Descriere", 332, HorizontalAlignment.Center);
        listView1.View = View.Details;
        SqlCommand cmd = new SqlCommand("select * from tiptaxe", con);
        SqlDataAdapter adp = new SqlDataAdapter(cmd);
        DataSet ds = new DataSet();
        adp.Fill(ds, "test");
        DataTable dt = ds.Tables["test"];
        for(int i=0;i<dt.Rows.Count;i++)
        {
            listView1.Items.Add(dt.Rows[i].ItemArray[0].ToString());
            listView1.Items[i].SubItems.Add(dt.Rows[i].ItemArray[1].ToString());
            listView1.Items[i].SubItems.Add(dt.Rows[i].ItemArray[2].ToString());
        }
        con.Close();
    }
    catch (Exception ee)
    {
        MessageBox.Show(ee.ToString());
    }
}

private void BATaxa_Click(object sender, EventArgs e)
{
    if(listView1.SelectedIndices.Count!=1)
    {
        MessageBox.Show("Nu ati selectat o taxa", "Eroare",
        MessageBoxButtons.OK);
        return;
    }
    if(!ValidNr(LElev.Text))
    {
        MessageBox.Show("Nu ati selectat un elev", "Eroare",
        MessageBoxButtons.OK);
        return;
    }
    if (!ValidNr(TBTaxa.Text))
    {
        MessageBox.Show("Valoare taxa invalida", "Eroare", MessageBoxButtons.OK);
        return;
    }
    DialogResult dr = MessageBox.Show($"Adaugati o taxa in valoare de
    {TBTaxa.Text} lei elevului cu id-ul {LElev.Text}", "Confirmare",
    MessageBoxButtons.YesNo);
    if(dr==DialogResult.Yes)
    {
        try
        {
            if (con.State != ConnectionState.Open)
                con.Open();

```

```

public partial class Rapoarte : Form
{
    public Rapoarte()
    {
        InitializeComponent();
    }
    readonly SqlConnection con = new SqlConnection(ClasaConString.ConString());
    private void BBack_Click(object sender, EventArgs e)
    {
        this.Close();
        Application.OpenForms["FereastraPrincipala"].Show();
    }
    private void Rapoarte_FormClosing(object sender, FormClosingEventArgs e)
    {
        if (new StackTrace().GetFrames().Any(x => x.GetMethod().Name == "Close"))
            return;
        else
            Application.OpenForms["FereastraPrincipala"].Close();
    }

    private void BElevi_Click(object sender, EventArgs e)
    {
        try
        {
            if (con.State != ConnectionState.Open)
                con.Open();
            LV.BackgroundImage = null;
            LV.Columns.Clear();
            LV.Items.Clear();
            LV.Columns.Add("Nume", 120, HorizontalAlignment.Center);
            LV.Columns.Add("Prenume", 120, HorizontalAlignment.Center);
            LV.Columns.Add("Clasa", 50, HorizontalAlignment.Center);
            LV.Columns.Add("Email", 120, HorizontalAlignment.Center);
            LV.Columns.Add("Nr. de Tlf.", 120, HorizontalAlignment.Center);
            LV.Columns.Add("Judet", 120, HorizontalAlignment.Center);
            LV.Columns.Add("Oras", 120, HorizontalAlignment.Center);
            LV.View = View.Details;
            SqlCommand cmd = new SqlCommand("select nume,prenume,clasa,"+
                " email,nrtlf,judet,oras from elevi order by nume asc", con);
            SqlDataAdapter adp = new SqlDataAdapter(cmd);
            DataSet ds = new DataSet();
            adp.Fill(ds, "test");
            DataTable dt = ds.Tables["test"];
            for (int i = 0; i < dt.Rows.Count; i++)
            {
                LV.Items.Add(dt.Rows[i].ItemArray[0].ToString());
                for(int j=1;j<=6;j++)
                    LV.Items[i].SubItems.Add(dt.Rows[i].ItemArray[j].ToString());
            }
            con.Close();
        }
        catch (Exception ee)
        {
            MessageBox.Show(ee.ToString());
        }
    }
}

```

```

private void BProf_Click(object sender, EventArgs e)
{
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        LV.BackgroundImage = null;
        LV.Columns.Clear();
        LV.Items.Clear();
        LV.Columns.Add("Nume", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Prenume", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Materia", 103, HorizontalAlignment.Center);
        LV.Columns.Add("Experienta", 80, HorizontalAlignment.Center);
        LV.Columns.Add("Email", 110, HorizontalAlignment.Center);
        LV.Columns.Add("Nr. de Tlf.", 100, HorizontalAlignment.Center);
        LV.Columns.Add("Judet", 100, HorizontalAlignment.Center);
        LV.Columns.Add("Oras", 100, HorizontalAlignment.Center);
        LV.View = View.Details;
        SqlCommand cmd = new SqlCommand("select nume,prenume,materie,experienta,"
+
            " email,nrtlf,judet,oras from profesori order by nume asc", con);
        SqlDataAdapter adp = new SqlDataAdapter(cmd);
        DataSet ds = new DataSet();
        adp.Fill(ds, "test");
        DataTable dt = ds.Tables["test"];
        for (int i = 0; i < dt.Rows.Count; i++)
        {
            LV.Items.Add(dt.Rows[i].ItemArray[0].ToString());
            for (int j = 1; j <= 7; j++)
                LV.Items[i].SubItems.Add(dt.Rows[i].ItemArray[j].ToString());
        }
        con.Close();
    }
    catch (Exception ee)
    {
        MessageBox.Show(ee.ToString());
    }
}

private void BTaxe_Click(object sender, EventArgs e)
{
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        LV.Columns.Clear();
        LV.Items.Clear();
        LV.BackgroundImage = null;
        LV.Columns.Add("Nume", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Prenume", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Email", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Nr. de Tlf.", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Taxa", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Valoare", 90, HorizontalAlignment.Center);
        LV.Columns.Add("Platit", 90, HorizontalAlignment.Center);
        LV.View = View.Details;
        SqlCommand cmd = new SqlCommand("select
elevi.nume,elevi.prenume,elevi.email,elevi.nrtlf,tiptaxe.denumire,taxe.valoare,taxe.plati

```

```

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;
using System.Data.SqlClient;
using System.Data.SqlTypes;
using System.Diagnostics;

namespace SistemDeEvidenta
{
    public partial class Profesorii : Form
    {
        public static string CONSTRING = ClasaConString.ConString();
        readonly SqlConnection con = new SqlConnection(CONSTRING);

        public Profesorii()
        {
            InitializeComponent();
        }

        private void BGolire_Click(object sender, EventArgs e)
        {
            RBMasc.Checked = RBFem.Checked = false;
            TBEmail.Text = TBTLf.Text = "";
            TBAdresa.Text = "";
            TBNuma.Text = TBPrename.Text = "";
            DatePickerNastere.Value = DateTime.Now;
            LProf.Text = "";
            CBExperienta.SelectedIndex = -1;
            try
            {
                PuneJudete();
                PuneMaterii();
            }
            catch (Exception ee)
            {
                MessageBox.Show(ee.ToString());
            }
        }

        private void BBack_Click(object sender, EventArgs e)
        {
            this.Close();
            Application.OpenForms["FereastraPrincipala"].Show();
        }

        private void Profesorii_FormClosing(object sender, FormClosingEventArgs e)
        {
            if (new StackTrace().GetFrames().Any(x => x.GetMethod().Name == "Close"))
                return;
            else
                Application.OpenForms["FereastraPrincipala"].Close();
        }
    }
}

```

```

namespace SistemDeEvidenta
{
    partial class Elevi
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            System.ComponentModel.ComponentResourceManager resources = new
System.ComponentModel.ComponentResourceManager(typeof(Elevi));
            this.panel1 = new System.Windows.Forms.Panel();
            this.panel3 = new System.Windows.Forms.Panel();
            this.panel4 = new System.Windows.Forms.Panel();
            this.panel12 = new System.Windows.Forms.Panel();
            this.DGVCriteriu = new System.Windows.Forms.DataGridView();
            this.panel5 = new System.Windows.Forms.Panel();
            this.panel7 = new System.Windows.Forms.Panel();
            this.BCautare = new System.Windows.Forms.Button();
            this.panel6 = new System.Windows.Forms.Panel();
            this.CBCriteriu = new System.Windows.Forms.ComboBox();
            this.TBCriteriu = new System.Windows.Forms.TextBox();
            this.BBack = new System.Windows.Forms.Button();
            this.CBORas = new System.Windows.Forms.ComboBox();
            this.CBJudet = new System.Windows.Forms.ComboBox();
            this.label12 = new System.Windows.Forms.Label();
            this.label13 = new System.Windows.Forms.Label();
            this.label11 = new System.Windows.Forms.Label();
            this.DateTimePickerInreg = new System.Windows.Forms.DateTimePicker();
            this.label10 = new System.Windows.Forms.Label();
            this.DateTimePickerNastere = new System.Windows.Forms.DateTimePicker();
            this.TBAdresa = new System.Windows.Forms.TextBox();
            this.label9 = new System.Windows.Forms.Label();
            this.TBTlf = new System.Windows.Forms.TextBox();
            this.TBEmail = new System.Windows.Forms.TextBox();
            this.label8 = new System.Windows.Forms.Label();
        }
    }
}

```

```

private void CBJudet_SelectionChangeCommitted(object sender, EventArgs e)
{
    string judet = CBJudet.Text;
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        string sqr = $"select id from judete where judet='{judet}'";
        SqlCommand cmd = new SqlCommand(sqr, con);
        SqlDataReader rdr = cmd.ExecuteReader();
        DataTable dt = new DataTable();
        dt.Columns.Add("id", typeof(int));
        dt.Load(rdr);
        int idjud = Int32.Parse(dt.Rows[0]["id"].ToString());

        sqr = $"select oras from orase where idjudet={idjud} order by oras asc";
        SqlCommand ncmd = new SqlCommand(sqr, con);
        SqlDataReader nrdr = ncmd.ExecuteReader();
        DataTable ndt = new DataTable();
        ndt.Columns.Add("oras", typeof(string));
        ndt.Load(nrdr);
        CBOoras.ValueMember = "oras";
        CBOoras.DataSource = ndt;
        con.Close();
        CBOoras.SelectedIndex = -1;
    }
    catch (Exception ee)
    {
        MessageBox.Show(ee.ToString());
    }
}

private void Bstergere_Click(object sender, EventArgs e)
{
    string text = LProf.Text;
    if (string.IsNullOrEmpty(text))
    {
        MessageBox.Show("Nu ati selectat niciun elev.", "Eroare",
        MessageBoxButtons.OK);
        return;
    }
    try
    {
        int id = Int16.Parse(text);
        DialogResult dr = MessageBox.Show($"Stergeti profesorul cu id-ul {id}?",
        "Confirmare", MessageBoxButtons.YesNo);
        if (dr == DialogResult.Yes)
        {
            if (con.State != ConnectionState.Open)
                con.Open();
            SqlCommand cmd = new SqlCommand($"delete from profesori where
            id={id}", con);
            cmd.ExecuteNonQuery();
            MessageBox.Show("Stergere reusita");
            BDate_Click(this, EventArgs.Empty);
            con.Close();
        }
    }
}

```

```

this.BCautare.BackColor = System.Drawing.Color.DarkKhaki;
this.BCautare.Dock = System.Windows.Forms.DockStyle.Fill;
this.BCautare.Font = new System.Drawing.Font("Nirmala UI", 8F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0)));
this.BCautare.ForeColor = System.Drawing.Color.Snow;
this.BCautare.Location = new System.Drawing.Point(0, 0);
this.BCautare.Margin = new System.Windows.Forms.Padding(0);
this.BCautare.Name = "BCautare";
this.BCautare.Size = new System.Drawing.Size(200, 32);
this.BCautare.TabIndex = 0;
this.BCautare.TabStop = false;
this.BCautare.Text = "Cautare";
this.BCautare.UseVisualStyleBackColor = false;
this.BCautare.Click += new System.EventHandler(this.BCautare_Click);
//
// panel6
//
this.panel6.Controls.Add(this.CBCriteriu);
this.panel6.Controls.Add(this.TBCriteriu);
this.panel6.Dock = System.Windows.Forms.DockStyle.Bottom;
this.panel6.Location = new System.Drawing.Point(0, 32);
this.panel6.Margin = new System.Windows.Forms.Padding(0);
this.panel6.Name = "panel6";
this.panel6.Size = new System.Drawing.Size(200, 80);
this.panel6.TabIndex = 0;
//
// CBCriteriu
//
this.CBCriteriu.Dock = System.Windows.Forms.DockStyle.Fill;
this.CBCriteriu.DropDownStyle =
System.Windows.Forms.ComboBoxStyle.DropDownList;
this.CBCriteriu.Font = new System.Drawing.Font("Nirmala UI", 8F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0)));
this.CBCriteriu.FormattingEnabled = true;
this.CBCriteriu.ItemHeight = 17;
this.CBCriteriu.Items.AddRange(new object[] {
"Nume",
"Prenume",
"Clasa",
"Nr. Tlf.",
"Email",
"Judet",
"Oras",
"Adresa",
"Sex"});
this.CBCriteriu.Location = new System.Drawing.Point(0, 0);
this.CBCriteriu.Margin = new System.Windows.Forms.Padding(0);
this.CBCriteriu.Name = "CBCriteriu";
this.CBCriteriu.Size = new System.Drawing.Size(200, 25);
this.CBCriteriu.TabIndex = 0;
this.CBCriteriu.TabStop = false;
//
// TBCriteriu
//
this.TBCriteriu.AccessibleDescription = "TBCriteriu";
this.TBCriteriu.Dock = System.Windows.Forms.DockStyle.Bottom;
this.TBCriteriu.Location = new System.Drawing.Point(0, 20);
this.TBCriteriu.Margin = new System.Windows.Forms.Padding(0);

```

```

partial class Admin
{
    /// <summary>
    /// Required designer variable.
    /// </summary>
    private System.ComponentModel.IContainer components = null;

    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
    protected override void Dispose(bool disposing)
    {
        if (disposing && (components != null))
        {
            components.Dispose();
        }
        base.Dispose(disposing);
    }

    #region Windows Form Designer generated code

    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        System.ComponentModel.ComponentResourceManager resources = new
System.ComponentModel.ComponentResourceManager(typeof(Admin));
        this.panel1 = new System.Windows.Forms.Panel();
        this.LV1 = new System.Windows.Forms.ListView();
        this.label5 = new System.Windows.Forms.Label();
        this.LV = new System.Windows.Forms.ListView();
        this.label4 = new System.Windows.Forms.Label();
        this.panel2 = new System.Windows.Forms.Panel();
        this.BAdd = new System.Windows.Forms.Button();
        this.TBDescriere = new System.Windows.Forms.TextBox();
        this.TBDenumire = new System.Windows.Forms.TextBox();
        this.label3 = new System.Windows.Forms.Label();
        this.label2 = new System.Windows.Forms.Label();
        this.label1 = new System.Windows.Forms.Label();
        this.BBack = new System.Windows.Forms.Button();
        this.BSterge = new System.Windows.Forms.Button();
        this.label6 = new System.Windows.Forms.Label();
        this.LabelD = new System.Windows.Forms.Label();
        this.panel1.SuspendLayout();
        this.panel2.SuspendLayout();
        this.SuspendLayout();
        //
        // panel1
        //
        this.panel1.BackColor = System.Drawing.Color.LightSkyBlue;
        this.panel1.Controls.Add(this.LV1);
        this.panel1.Controls.Add(this.label5);
        this.panel1.Controls.Add(this.LV);
        this.panel1.Controls.Add(this.label4);
    }
}

```



```

private void CBJudet_SelectionChangeCommitted(object sender, EventArgs e)
{
    string judet = CBJudet.Text;
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        string sqr = $"select id from judete where judet='{judet}'";
        SqlCommand cmd = new SqlCommand(sqr, con);
        SqlDataReader rdr = cmd.ExecuteReader();
        DataTable dt = new DataTable();
        dt.Columns.Add("id", typeof(int));
        dt.Load(rdr);
        int idjud = Int32.Parse(dt.Rows[0]["id"].ToString());

        sqr = $"select oras from orase where idjudet={idjud} order by oras asc";
        SqlCommand ncmd = new SqlCommand(sqr, con);
        SqlDataReader nrdr = ncmd.ExecuteReader();
        DataTable ndt = new DataTable();
        ndt.Columns.Add("oras", typeof(string));
        ndt.Load(nrdr);
        CBOoras.ValueMember = "oras";
        CBOoras.DataSource = ndt;
        con.Close();
        CBOoras.SelectedIndex = -1;
    }
    catch (Exception ee)
    {
        MessageBox.Show(ee.ToString());
    }
}

private void Bstergere_Click(object sender, EventArgs e)
{
    string text = LProf.Text;
    if (string.IsNullOrEmpty(text))
    {
        MessageBox.Show("Nu ati selectat niciun elev.", "Eroare",
        MessageBoxButtons.OK);
        return;
    }
    try
    {
        int id = Int16.Parse(text);
        DialogResult dr = MessageBox.Show($"Stergeti profesorul cu id-ul {id}?",
        "Confirmare", MessageBoxButtons.YesNo);
        if (dr == DialogResult.Yes)
        {
            if (con.State != ConnectionState.Open)
                con.Open();
            SqlCommand cmd = new SqlCommand($"delete from profesori where
            id={id}", con);
            cmd.ExecuteNonQuery();
            MessageBox.Show("Stergere reusita");
            BDate_Click(this, EventArgs.Empty);
            con.Close();
        }
    }
}

```

```

private void BAdaugare_Click(object sender, EventArgs e)
{
    bool ok = ValidareDate();
    if (!ok) return;
    try
    {
        if (con.State != ConnectionState.Open)
            con.Open();
        string sqr = "insert into profesori values(" +
            "@materie," +
            "@sex," +
            "@nume," +
            "@prenume," +
            "@email," +
            "@nrtrlf," +
            "@adresa," +
            "@datanast," +
            "@experienta," +
            "@judet," +
            "@idjud," +
            "@oras," +
            "@idoras" +
            ")";

        SqlCommand cmd;
        try
        {
            cmd = new SqlCommand(sqr, con);
        }
        catch (Exception es)
        {
            MessageBox.Show(es.ToString());
            return;
        }
        string materie = CBMaterii.Text;
        cmd.Parameters.AddWithValue("@materie", materie);

        bool sex;
        if (RBMasc.Checked == true)
            sex = true;
        else sex = false;
        cmd.Parameters.AddWithValue("@sex", sex);

        string nume = TBNume.Text;
        cmd.Parameters.AddWithValue("@nume", nume);

        string prenume = TBPrenume.Text;
        cmd.Parameters.AddWithValue("@prenume", prenume);

        string email = TBEmail.Text;
        cmd.Parameters.AddWithValue("@email", email);

        string nrtrlf = TBTlrf.Text;
        cmd.Parameters.AddWithValue("@nrtrlf", nrtrlf);

        string adresa = TBAdresa.Text;
        cmd.Parameters.AddWithValue("@adresa", adresa);
    }
}

```

```

private void BBack_Click(object sender, EventArgs e)
{
    this.Close();
    Application.OpenForms["FereastraPrincipala"].Show();
}

private void Profesori_FormClosing(object sender, FormClosingEventArgs e)
{
    if (new StackTrace().GetFrames().Any(x => x.GetMethod().Name == "Close"))
        return;
    else
        Application.OpenForms["FereastraPrincipala"].Close();
}

public void PuneJudete()
{
    if (con.State != ConnectionState.Open)
        con.Open();
    SqlCommand cmd = new SqlCommand("select judet from judete order by judet
asc", con);
    SqlDataReader rdr = cmd.ExecuteReader();
    DataTable dt = new DataTable();
    dt.Columns.Add("judet", typeof(string));
    dt.Load(rdr);
    CBJudet.ValueMember = "judet";
    CBJudet.DataSource = dt;
    con.Close();
    CBJudet_SelectionChangeCommitted(this, EventArgs.Empty);
    CBJudet.SelectedIndex = -1;
}

public void PuneMaterii()
{
    if (con.State != ConnectionState.Open)
        con.Open();
    SqlCommand cmd = new SqlCommand("select materie from materii order by materie
asc", con);
    SqlDataReader rdr = cmd.ExecuteReader();
    DataTable dt = new DataTable();
    dt.Columns.Add("materie", typeof(string));
    dt.Load(rdr);
    CBMaterii.ValueMember = "materie";
    CBMaterii.DataSource = dt;
    con.Close();
    CBMaterii.SelectedIndex = -1;
}

```

## 6. Bibliografie

- <https://www.w3schools.com/cs/index.php>
- <https://docs.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual-studio?view=vs-2022>
- <https://www.c-sharpcorner.com/article/windows-form-application-c-sharp/>
- <https://www.w3schools.com/sql/>
- [https://www.tutorialspoint.com/ms\\_sql\\_server/ms\\_sql\\_server\\_management\\_studio.htm](https://www.tutorialspoint.com/ms_sql_server/ms_sql_server_management_studio.htm)