

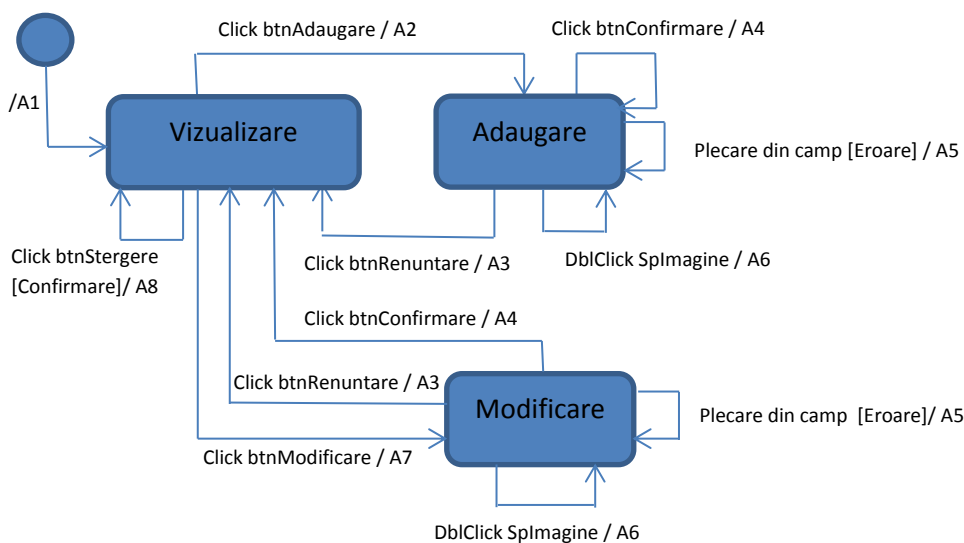
## TEHNICI DE PROGRAMARE AVANSATA – Laborator 9

### 1. Implementarea unui model avansat de vizualizare/actualizare date dintr-o tabela

Modelul prezentat in continuare este un model mai complex pentru vizualizare/actualizare tabela, ce functioneaza in urmatoarele conditii:

1. Actualizarea nu se mai face direct in grid ci in componente distincte pentru fiecare camp in parte;
2. Vom avea butoane distincte pentru Adaugare, Modificare si Stergere
3. Restrictiile de integritate (duplicat pe cheie semantica, stergere entitate referita in alta tabela) nu mai trebuiesc implementate la nivelul bazei de date, intru-cat sunt tratate in aplicatie.
4. Tabela poate avea oricate chei externe, deci legaturi cu alte tabele.
5. Putem atasa inregistrarilor fisiere imagine
6. Putem atasa campurilor componente specializate pentru introducerea datelor (DateTime Picker, OpenFileDialog, ComboBox, etc)
7. Validarile (tip de date, duplicat pe cheie semantica, stergere inregistrare referita, etc) pot fi facute imediat dupa ce a fost parasit campul.

Modelul pentru vizualizare/actualizare este prezentat in diagrama de stare urmatoare:



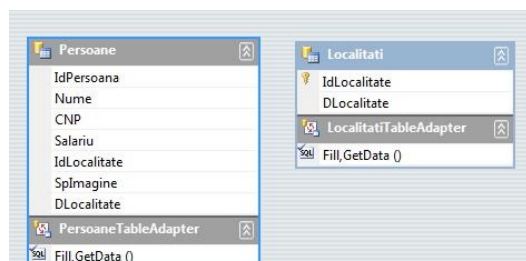
**Studiu de caz:** sa consideram o baza de date cu 3 tabele:

Persoane ( IdPersoana, Nume, CNP, Salariu, IdLocalitate, SplImagine)

Localitati (IdLocalitate, DLocalitate)

Auto(IdAuto, NrInmatriculare, Marca, Model, CapacitateMotor, AnFabricatie, IdProprietar)

Structura DataSet-ului:



## Structura Form-ului:

The screenshot shows a Windows Form titled "Form2". At the top, there is a menu bar with five buttons: "Adaugare", "Modificare", "Stergere", "Confirmare", and "Renuntare". Below the menu bar, on the left, is a table with three columns: "IdPersoana", "Nume", and "CNP". The table has a header row with a "\*" in the first column and a data row. To the right of the table is a form with several input fields: "Nume", "CNP", "Salariu", "Localitate" (a dropdown menu), "Poza", and "ID". A dashed box encloses the "CNP", "Salariu", "Localitate", "Poza", and "ID" fields. Below the form, there is a toolbar with icons for "openFileDialog1", "dataSet1", "persoaneBindingSource", "persoaneTableAdapter", "localitatiTableAdapter", and "localitatiBindingSource".

```
private void FPersoane_Load(object sender, EventArgs e)
{
    A1();
}

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

private void btnRenuntare_Click(object sender, EventArgs e)
{
    A3();
}

private void btnConfirmare_Click(object sender, EventArgs e)
{
    A4();
}

private void txtSalariu_Leave(object sender, EventArgs e)
{
    A5(txtSalariu) ;
}

private void txtCNP_Leave(object sender, EventArgs e)
{
    A5(txtCNP);
}

private void txtSpImagine_MouseDoubleClick(object sender, MouseEventArgs e)
{
    A6();
}

private void btnModificare_Click(object sender, EventArgs e)
{
    A7();
}

private void btnStergere_Click(object sender, EventArgs e)
{
    const string mesaj = "Confirmati stergerea";
    const string titlu = "Stergere inregistrare";
    var rezultat = MessageBox.Show(mesaj, titlu, MessageBoxButtons.YesNo,
```

```

        MessageBoxIcon.Warning);
    if (rezultat == DialogResult.No) return;
    A8();
}

private void A1(){

    //Umple cu date obiectele DataTable: Persoane, Localitati (din DataSet):
    persoaneTableAdapter.Fill(dataSet1.Persoane);
    localitatiTableAdapter.Fill(dataSet1.Localitati);

    //Configurare comboBox-uri
    cmbLocalitate.DropDownStyle = ComboBoxStyle.Simple;

    //Configurare PB
    PB.SizeMode = PictureBoxSizeMode.StretchImage;

    //Protectie grid
    dataGridView1.AllowUserToAddRows = false;
    dataGridView1.AllowUserToDeleteRows = false;
    dataGridView1.ReadOnly = true;

    //Protectie txtIdPersoana
    txtIdPersoana.ReadOnly = true;

    A3();
}

private void A2()
{
    //Configurare butoane
    configureazaButoane(false);

    //Dezlegare controale Panel
    legareControale(false);

    //Ridicare protectie controale Panel
    protectiePanel(false);

    //Configurare(ComboBox - Uri)
    cmbLocalitate.DropDownStyle = ComboBoxStyle.DropDownList;

    //Legare comboBox-uri la sursa de date
    cmbLocalitate.DataSource = localitatiBindingSource;
    cmbLocalitate.DisplayMember = "DLocalitate";
    cmbLocalitate.ValueMember = "IdLocalitate";

    //Modificare lblOp
    lblOp.Text = "ADAUGARE";

    //Pozitionare cursor pe primul camp
    txtNume.Focus();

    // Golire campuri
    golireCampuri();
}

private void A3()
{
    //Configurare comboBox-uri
    cmbLocalitate.DropDownStyle = ComboBoxStyle.Simple;

    //Initializare lblOp
    lblOp.Text = "";

    //Configurare(butoane)
    configureazaButoane(true);
}

```

```

        //Protectie componente Panel
        protectiePanel(true);

        //Legare controale
        legareControale(true);
    }

    private void A4()
    {
        if (lblOp.Text == "ADAUGARE")
        {
            if ( ! validareCampuriObligatorii() ) return;

            adauga_inregistrare();

            golireCampuri();

            //Pune cursor pe primul camp
            txtNume.Focus();
            refresh_grid(persoaneBindingSource.Position);
        }
        else if (lblOp.Text == "MODIFICARE")
        {
            modifica_inregistrare();
            refresh_grid(persoaneBindingSource.Position);
            A3();
        }
        else
            MessageBox.Show("Operatie actualizare neefectuata");
    }

    private void A5(TextBox txtB)
    {
        decimal p;

        if (txtB.Text == "") return;
        try
        {
            p = Convert.ToDecimal(txtB.Text);
        }
        catch
        {
            MessageBox.Show("Format eronat");
            txtB.Focus();
        }
    }

    private void A6()
    {
        if (lblOp.Text == "")
            return;

        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            txtSpImagine.Text = openFileDialog1.FileName;
            PB.ImageLocation = txtSpImagine.Text;
        }
    }

    private void A7()
    {
        //Configurare butoane
        configureazaButoane(false);

        //Dezlegare controale Panel
        legareControale(false);

        //Ridicare protectie controale Panel
    }

```

```

        protectiePanel(false);

        //Configurare(ComboBox - Uri)
        cmbLocalitate.DropDownStyle = ComboBoxStyle.DropDownList;

        //Legare comboBox-uri la sursa de date
        cmbLocalitate.DataSource = localitatiBindingSource;
        cmbLocalitate.DisplayMember = "DLocalitate";
        cmbLocalitate.ValueMember = "IdLocalitate";

        //Modificare lblOp
        lblOp.Text = "MODIFICARE";

        //Pozitionare cursor pe primul camp
        txtNume.Focus();
    }

    private void A8()
    {
        OleDbConnection con = new OleDbConnection();
        OleDbCommand cmd = new OleDbCommand();

        con.ConnectionString = persoaneTableAdapter.Connection.ConnectionString;

        cmd.Connection = con;

        cmd.CommandText = "Delete From Persoane Where IdPersoana = " + txtIdPersoana.Text;

        MessageBox.Show(cmd.CommandText);

        con.Open();

        cmd.ExecuteNonQuery();

        con.Close();

        refresh_grid(persoaneBindingSource.Position);
    }

    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)
        {
            txtNume.DataBindings.Add("Text", persoaneBindingSource, "Nume");
            txtCNP.DataBindings.Add("Text", persoaneBindingSource, "CNP");
            txtSalariu.DataBindings.Add("Text", persoaneBindingSource, "Salariu");
            cmbLocalitate.DataBindings.Add("Text", persoaneBindingSource, "DLocalitate");
            txtSpImagine.DataBindings.Add("Text", persoaneBindingSource, "SpImagine");
            txtIdPersoana.DataBindings.Add("Text", persoaneBindingSource, "IdPersoana");
            PB.DataBindings.Add("ImageLocation", persoaneBindingSource, "SpImagine");
        }
        else
        {
            txtNume.DataBindings.Clear();
            txtCNP.DataBindings.Clear();
            txtSalariu.DataBindings.Clear();
            cmbLocalitate.DataBindings.Clear();
            txtSpImagine.DataBindings.Clear();
        }
    }

```

```

        txtIdPersoana.DataBindings.Clear();
        PB.DataBindings.Clear();
    }
}

private void protectiePanel(bool v)
{
    txtNume.ReadOnly = v;
    txtCNP.ReadOnly = v;
    txtSalariu.ReadOnly = v;
    txtSpImagine.ReadOnly = v;
    cmbLocalitate.Enabled = ! v;
}

private void golireCampuri()
{
    txtNume.Text = "";
    txtCNP.Text = "";
    txtIdPersoana.Text = "";
    txtSalariu.Text = "";
    txtSpImagine.Text = "";
    cmbLocalitate.SelectedIndex = -1;
    PB.ImageLocation = "";
}

private bool validareCampuriObligatorii()
{
    //Validare de completare obligatorie campurile: Nume, CNP, Salariu, Localitate
    if (txtNume.Text == "")
    {
        MessageBox.Show("Completati Nume !");
        txtNume.Focus();
        return false;
    }
    if (txtCNP.Text == "")
    {
        MessageBox.Show("Completati CNP !");
        txtCNP.Focus();
        return false;
    }
    if (txtSalariu.Text == "")
    {
        MessageBox.Show("Completati Salariu !");
        txtSalariu.Focus();
        return false;
    }

    if (cmbLocalitate.Text == "")
    {
        MessageBox.Show("Completati Localitate !");
        cmbLocalitate.Focus();
        return false;
    }
    return true;
}

private void adauga_inregistrare()
{
    string listaCampuri;
    string listaValori;

    OleDbConnection con = new OleDbConnection();
    OleDbCommand cmd = new OleDbCommand();

    con.ConnectionString = persoaneTableAdapter.Connection.ConnectionString;

    cmd.Connection = con;

```

```

listaCampuri = "Nume, CNP, Salariu, IdLocalitate, SpImagine";
listaValori = "'" + txtNume.Text + "'" +
              "','" + txtCNP.Text + "'" +
              "','" + txtSalariu.Text +
              "','" + cmbLocalitate.SelectedValue +
              "','" + txtSpImagine.Text + "'";

cmd.CommandText = "Insert into Persoane(" + listaCampuri + ") " +
                  "Select " + listaValori;

MessageBox.Show(cmd.CommandText);

con.Open();

cmd.ExecuteNonQuery();

con.Close();
}

private void refresh_grid(int p)
{
    persoaneTableAdapter.Fill(dataSet1.Persoane);
    persoaneBindingSource.Position = p;
}

private void modifica_inregistrare()
{
    string listaSet;

    OleDbConnection con = new OleDbConnection();
    OleDbCommand cmd = new OleDbCommand();

    con.ConnectionString = persoaneTableAdapter.Connection.ConnectionString;

    cmd.Connection = con;

    listaSet = "Nume = '" + txtNume.Text + "','" +
              "CNP = '" + txtCNP.Text + "','" +
              "Salariu = '" + txtSalariu.Text + "','" +
              "IdLocalitate = '" + cmbLocalitate.SelectedValue + "','" +
              "SpImagine = '" + txtSpImagine.Text + "'";

    cmd.CommandText = "Update Persoane Set " + listaSet + " Where IdPersoana=" +
                      txtIdPersoana.Text;

    MessageBox.Show(cmd.CommandText);

    con.Open();

    cmd.ExecuteNonQuery();

    con.Close();
}

```