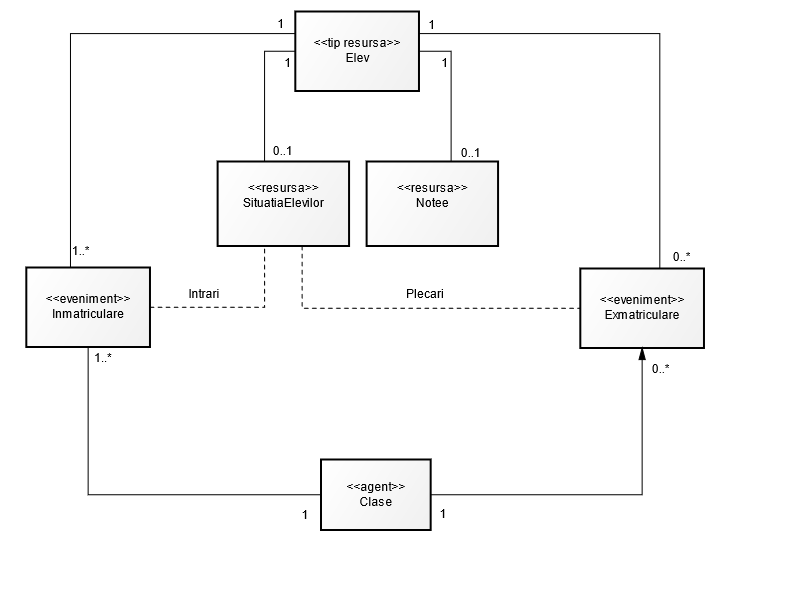
**6.Evidenta elevilor dintr-un liceu**

Platon Mihai-Valentin, INFO3, Grupa 3



**Tabele:**

* **Elev:**

IdElev – AutoNumber PK

Nume - ShortText(255) Required

InitialaTata – ShortText(255) Required

Prenume – ShortText(255) Required

Clasa – ShortText(5) Required

Adresa – ShortText(255) Required

* **Inmatriculari:**

IdInmatriculare – AutoNumber PK

IdElev – Number FK

DataInmatricularii – Date (short)

* **Exmatriculari**:

IdElev – Number FK

NumeComplet - ShortText(255)

Motiv – ShortText(255)

DataExmatricularii – Date (short)

* **Clase:**

IdClasa – AutoNumber PK

DenumireClasa – ShortText(5)

* **SituatieElev:**

IdSituatie – AutoNumber PK

IdElev – Number FK

IdClasa – Number FK

* **Notee:**

IdElev – Number FK

\*Materii – 11 coloane, ShortText(255)

**Form-uri:**

**FStart**

**Graphical user interface, application, Teams

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

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 ProiectMPP

{

public partial class FStart : Form

{

private OleDbConnection con = new OleDbConnection();

private OleDbCommand cmd = new OleDbCommand();

private OleDbDataReader rdr;

public FStart()

{

InitializeComponent();

}

private void A1(Boolean v)

{

menuStrip1.Visible = !v;

lblTitlu.Visible = !v;

lblAutor.Visible = !v;

lblParola.Visible = v;

txtParola.Visible = v;

lblUtilizator.Visible = v;

txtUtilizator.Visible = v;

if (v) btnStart.Text = "Log In";

else btnStart.Text = "Log Out";

}

private Boolean Logare\_OK()

{

if (txtUtilizator.Text == "")

{

MessageBox.Show("Completati utilizator !");

txtUtilizator.Focus();

return false;

}

if (txtParola.Text == "")

{

MessageBox.Show("Completati parola !");

txtParola.Focus();

return false;

}

con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=C:\\Users\\Garry\\Desktop\\ProiectMPP.mdb";

cmd.Connection = con;

cmd.CommandText = "Select IdUtilizator,Parola from Utilizatori " +

"where Nume='" + txtUtilizator.Text + "'";

con.Open();

rdr = cmd.ExecuteReader();

if (rdr.Read())

{

if (txtParola.Text != rdr.GetString(1))

{

MessageBox.Show("Parola eronata");

txtParola.Focus();

con.Close();

return false;

}

con.Close();

return true;

}

else

{

MessageBox.Show("Utilizator eronat");

txtUtilizator.Focus();

con.Close();

return false;

}

}

private void btnStart\_Click(object sender, EventArgs e)

{

if (btnStart.Text == "Log In")

{

if (Logare\_OK()) A1(false);

}

else A1(true);

}

private void FStart\_Load(object sender, EventArgs e)

{

A1(true);

}

private void claseToolStripMenuItem\_Click(object sender, EventArgs e)

{

Clase f = new Clase();

f.ShowDialog();

}

private void noteToolStripMenuItem\_Click(object sender, EventArgs e)

{

Note f = new Note();

f.ShowDialog();

}

private void mediiToolStripMenuItem\_Click(object sender, EventArgs e)

{

Medii f = new Medii();

f.ShowDialog();

}

private void exmatriculareToolStripMenuItem\_Click(object sender, EventArgs e)

{

lblData f = new lblData();

f.ShowDialog();

}

}

}

**Clase**

**Chart

Description automatically generated**

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;

using System.Windows.Forms.DataVisualization.Charting;

namespace ProiectMPP

{

public partial class Clase : Form

{

private int nrElevi, nrClase;

OleDbConnection con = new OleDbConnection();

OleDbCommand cmd = new OleDbCommand();

OleDbDataReader r;

private string lastCell;

public Clase()

{

InitializeComponent();

}

private void Clase\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'claseDS.Clasa' table. You can move, or remove it, as needed.

this.clasaTableAdapter.Fill(this.claseDS.Clasa);

// TODO: This line of code loads data into the 'claseDS.Elev' table. You can move, or remove it, as needed.

this.elevTableAdapter.Fill(this.claseDS.Elev);

this.dataGridView1.Sort(this.dataGridView1.Columns["Nume"], ListSortDirection.Ascending);

nrElevi = claseDS.Elev.Rows.Count;

nrClase = claseDS.Clasa.Rows.Count;

dataGridView1.Focus();

RefreshChart();

}

private void RefreshChart()

{

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

con.Open();

cmd.CommandText = "SELECT COUNT(\*) FROM Clase";

Int32 nrClase = (Int32)cmd.ExecuteScalar();

con.Close();

string[] xval = new string[nrClase ];

int[] yval = new int[nrClase ];

con.Close();

cmd.CommandText = "Select DenumireClasa From Clase ";

con.Open();

r = cmd.ExecuteReader();

int x = 0;

while (r.Read())

{

xval[x] = r.GetString(0);

x++;

}

foreach (DataRow r in claseDS.Elev)

{

con.Close();

string clasa = r["Clasa"].ToString();

int id = (int)r["IdElev"];

cmd.CommandText = "Select IdClasa From SituatieElev Where IdElev = " +id;

con.Open();

int idClasa = (int)cmd.ExecuteScalar();

con.Close();

yval[idClasa - 1]++;

}

chart1.ChartAreas["ChartArea1"].AxisX.Interval = 1;

chart1.ChartAreas[0].AxisY.Maximum = nrElevi;

chart1.Series[0].Points.DataBindXY(xval, yval);

chart1.Series[0].Name = "Grad ocupare";

chart1.Series[0].ChartType = SeriesChartType.Column;

}

private void cbClasa\_SelectedValueChanged(object sender, EventArgs e)

{

if (dataGridView1.Enabled)

{

string clasaAleasa = cbClasa.Text;

elevBindingSource.Filter = "Clasa = " + "'" + clasaAleasa + "'";

this.dataGridView1.Sort(this.dataGridView1.Columns["Nume"], ListSortDirection.Ascending);

dataGridView1.Focus();

}

}

private void btnAdaugare\_Click(object sender, EventArgs e)

{

A1(true);

A3(true);

lblMain.Text = "ADAUGARE";

txtNume.Focus();

txtNume.Text = null;

txtInitialaTatalui.Text = null;

txtPrenume.Text = null;

cmbClasaPanel.SelectedItem = null;

txtLocalitate.Text = null;

txtAdresa.Text = null;

dataGridView1.Enabled = false;

cbClasa.Enabled = false;

}

private void btnModificare\_Click(object sender, EventArgs e)

{

A1(true);

lblMain.Text = "MODIFICARE/TRANSFER";

A3(false);

}

private void btnConfirmare\_Click(object sender, EventArgs e)

{

if (lblMain.Text == "ADAUGARE")

{

dateTimePicker1.Enabled = true;

if (txtNume.Text == "")

{

MessageBox.Show("Numele nu a fost adaugat!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (txtPrenume.Text == "")

{

MessageBox.Show("Prenumele nu a fost adaugat!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if(cmbClasaPanel.SelectedIndex == -1)

{

MessageBox.Show("Clasa nu a fost selectata!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (txtLocalitate.Text == "")

{

MessageBox.Show("Localitatea nu a fost adaugata!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (txtAdresa.Text == "")

{

MessageBox.Show("Adresa nu a fost adaugata!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

string listaCampuri = "Nume, InitialaTata, Prenume, Clasa, Localitate, Adresa";

string listaValori =

"'" + txtNume.Text + "'" +

",'" + txtInitialaTatalui.Text + "'" +

",'" + txtPrenume.Text + "'" +

",'" + cmbClasaPanel.Text + "'" +

",'" + txtLocalitate.Text + "'" +

",'" + txtAdresa.Text + "'";

cmd.CommandText = "Insert into Elev(" + listaCampuri + ") " +

"Values " + "(" + listaValori + ")";

con.Open();

cmd.ExecuteNonQuery();

con.Close();

elevTableAdapter.Update(claseDS.Elev);

cmd.CommandText = "SELECT TOP 1 IdElev FROM Elev ORDER BY IdElev DESC";

con.Open();

int ultimulId = (int)cmd.ExecuteScalar();

con.Close();

cmd.CommandText = "Select IdClasa From Clase Where DenumireClasa = " + "'" + cmbClasaPanel.Text + "'";

con.Open();

int clasaId = (int)cmd.ExecuteScalar();

con.Close();

cmd.CommandText = "INSERT INTO SituatieElev(IdElev, IdClasa) Values(" + ultimulId +", " + clasaId +")";

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "INSERT INTO Inmatriculari(IdElev, DataInmatriculare) Values(" + ultimulId + ", '"

+ dateTimePicker1.Value.ToShortDateString() + "')";

con.Open();

cmd.ExecuteNonQuery();

con.Close();

}

else if(lblMain.Text == "MODIFICARE/TRANSFER")

{

dateTimePicker1.Enabled = false;

int i = dataGridView1.CurrentCell.RowIndex;

string lastCell = dataGridView1.Rows[i].Cells[0].Value.ToString();

int lastCellInt = int.Parse(lastCell);

string clasa = dataGridView1.Rows[i].Cells[4].Value.ToString();

var clasaString = clasa;

var clasaStringCommands = clasaString.Split(new[] { ' ' }, 2);

var clasaNr = clasaStringCommands[0];

var clasaSimbol = clasaStringCommands[1];

string clasaNoua;

if (txtNume.Text == "")

{

MessageBox.Show("Numele este gol!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (txtPrenume.Text == "")

{

MessageBox.Show("Prenumele este gol!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (txtLocalitate.Text == "")

{

MessageBox.Show("Localitatea lipseste!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (txtAdresa.Text == "")

{

MessageBox.Show("Adresa lipseste!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

con.Open();

string listaSet =

"Nume = '" + txtNume.Text + "'" +

", InitialaTata = '" + txtInitialaTatalui.Text + "'" +

", Prenume = '" + txtPrenume.Text + "'" +

", Clasa = '" + clasaNr.ToString() + " " + cmbClasaPanel.Text + "'" +

", Localitate ='" + txtLocalitate.Text + "'" +

", Adresa = '" + txtAdresa.Text + "'";

lastCellInt = int.Parse(lastCell);

cmd.CommandText = "Update Elev Set " + listaSet + " Where IdElev=" + lastCellInt;

cmd.ExecuteNonQuery();

con.Close();

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.CommandText = "SELECT IdClasa FROM Clase Where DenumireClasa = '"

+ clasaNr.ToString() + " " + cmbClasaPanel.Text + "'";

Int32 idClasa = (Int32)cmd.ExecuteScalar();

cmd.CommandText = "Update SituatieElev Set IdClasa = " + idClasa + " Where IdElev = " + lastCellInt;

cmd.ExecuteNonQuery();

con.Close();

}

else

{

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

if(dataGridView1.Rows.Count == 1)

{

lastCell = dataGridView1.Rows[0].Cells[0].Value.ToString();

}

else

{

lastCell = dataGridView1.SelectedRows[0].Cells[0].Value.ToString();

}

int lastCellInt = int.Parse(lastCell);

cmd.CommandText = "DELETE FROM Elev Where IdElev= " + lastCellInt;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "DELETE FROM Notee Where IdElev= " + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

}

A1(false);

A2(false);

elevTableAdapter.Fill(claseDS.Elev);

RefreshChart();

dataGridView1.Enabled = true;

cbClasa.Enabled = true;

lblMain.Text = "";

}

private void btnRenuntare\_Click(object sender, EventArgs e)

{

A1(false);

A2(false);

lblMain.Text = "";

dataGridView1.Enabled = true;

cbClasa.Enabled = true;

}

private void dataGridView1\_SelectionChanged(object sender, EventArgs e)

{

if (dataGridView1.Rows.Count > 0)

{

int i;

i = dataGridView1.CurrentCell.RowIndex;

txtNume.Text = dataGridView1.Rows[i].Cells[1].Value.ToString();

txtInitialaTatalui.Text = dataGridView1.Rows[i].Cells[2].Value.ToString();

txtPrenume.Text = dataGridView1.Rows[i].Cells[3].Value.ToString();

cmbClasaPanel.SelectedItem = dataGridView1.Rows[i].Cells[4].Value.ToString();

txtLocalitate.Text = dataGridView1.Rows[i].Cells[5].Value.ToString();

txtAdresa.Text = dataGridView1.Rows[i].Cells[6].Value.ToString();

}

}

private void btnExmatriculare\_Click(object sender, EventArgs e)

{

A1(true);

lblMain.Text = "EXMATRICULARE";

A2(true);

}

private void A1(bool v)

{

btnConfirmare.Enabled = v;

btnRenuntare.Enabled = v;

panel1.Visible = v;

btnAdaugare.Enabled = !v;

btnModificare.Enabled = !v;

//btnExmatriculare.Enabled = !v;

}

private void A2(bool v)

{

txtNume.ReadOnly = v;

txtInitialaTatalui.ReadOnly = v;

txtPrenume.ReadOnly = v;

txtAdresa.ReadOnly = v;

txtLocalitate.ReadOnly = v;

cmbClasaPanel.Enabled = !v;

}

private void A3(bool v)

{

cmbClasaPanel.Items.Clear();

if (v)

{

cmd.CommandText = "Select DenumireClasa From Clase ";

con.Open();

r = cmd.ExecuteReader();

while (r.Read())

{

cmbClasaPanel.Items.Add(r.GetString(0));

}

con.Close();

}

else

{

cmbClasaPanel.Items.Add("A");

cmbClasaPanel.Items.Add("B");

cmbClasaPanel.Items.Add("C");

}

}

}

}

**Note**

**Graphical user interface, application

Description automatically generated**

using System;

using System.Data.OleDb;

using System.Windows.Forms;

using System.Data;

using System.ComponentModel;

using System.Text.RegularExpressions;

namespace ProiectMPP

{

public partial class Note : Form

{

private OleDbConnection con = new OleDbConnection();

private OleDbCommand cmd = new OleDbCommand();

OleDbDataReader r;

private string lastCell;

public Note()

{

InitializeComponent();

}

private void Note\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'noteDS1.Notee' table. You can move, or remove it, as needed.

this.noteeTableAdapter.Fill(this.noteDS1.Notee);

// TODO: This line of code loads data into the 'noteDS.Note' table. You can move, or remove it, as needed.

this.noteeTableAdapter.Fill(this.noteDS.Notee);

// TODO: This line of code loads data into the 'noteDS.Elev' table. You can move, or remove it, as needed.

this.elevTableAdapter.Fill(this.noteDS.Elev);

// TODO: This line of code loads data into the 'noteDS.Note' table. You can move, or remove it, as needed.

// TODO: This line of code loads data into the 'noteDS.Note' table. You can move, or remove it, as needed.

this.noteeTableAdapter.Fill(this.noteDS.Notee);

// TODO: This line of code loads data into the 'noteDS.Elev' table. You can move, or remove it, as needed.

this.elevTableAdapter.Fill(this.noteDS.Elev);

this.dataGridView1.Sort(this.dataGridView1.Columns["Nume"], ListSortDirection.Ascending);

lastCell = dataGridView1.Rows[0].Cells[0].Value.ToString();

noteeBindingSource.Filter = "IdElev = " + lastCell;

A3();

}

private void cbClasa\_ValueMemberChanged(object sender, EventArgs e)

{

string clasaAleasa = cbClasa.Text;

elevBindingSource.Filter = "Clasa = " + "'" + clasaAleasa + "'";

this.dataGridView1.Sort(this.dataGridView1.Columns["Nume"], ListSortDirection.Ascending);

dataGridView1.Focus();

if (dataGridView1.Rows.Count > 0)

{

lastCell = dataGridView1.Rows[0].Cells[0].Value.ToString();

noteeBindingSource.Filter = "IdElev = " + lastCell;

}

}

private void elevBindingSource\_CurrentChanged(object sender, EventArgs e)

{

noteeTableAdapter.Fill(noteDS1.Notee);

if (dataGridView1.SelectedRows.Count > 0)

{

if (dataGridView1.SelectedRows != null && dataGridView1.SelectedRows[0].Cells[0].Value != null)

lastCell = dataGridView1.SelectedRows[0].Cells[0].Value.ToString();

noteeBindingSource.Filter = "IdElev = " + lastCell;

}

}

private void A1(bool v)

{

btnAdaugareNote.Enabled = !v;

btnMedie.Enabled = !v;

LblMaterie.Visible = v;

cmbMaterie.Visible = v;

LblNota.Visible = v;

cmbNota.Visible = v;

btnConfirmare.Visible = v;

btnRenuntare.Visible = v;

}

private void btnModificareNote\_Click(object sender, EventArgs e)

{

A1(true);

}

private void btnConfirmare\_Click(object sender, EventArgs e)

{

if(cmbMaterie.SelectedIndex == -1)

{

MessageBox.Show("Materia nu a fost selectata!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (cmbNota.SelectedIndex == -1)

{

MessageBox.Show("Nota nu a fost selectata!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

string materie = cmbMaterie.Text;

string nota = cmbNota.Text;

con.ConnectionString = noteeTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

//cmd.CommandText = "Insert into Note (" + materie + ")" + " Values (" + "'" + nota + "'" + ")" + " WHERE idElev =" + lastCell;

cmd.CommandText = "SELECT 1 FROM Notee WHERE IdElev =" + lastCell;

con.Open();

var reader = cmd.ExecuteReader();

if (!reader.Read())

{

con.Close();

cmd.CommandText = " Insert INTO Notee (IdElev ," + materie + ") Values(" + lastCell + "," + "'" + nota + "'" + ")";

con.Open();

cmd.ExecuteNonQuery();

con.Close();

}

else

{

con.Close();

cmd.CommandText = "Select " + materie + " From Notee Where IdElev = " + lastCell;

con.Open();

var reader2 = cmd.ExecuteScalar();

string readerValue = Convert.ToString(reader2);

if (!readerValue.Equals(""))

{

con.Close();

con.Open();

cmd.CommandText = "UPDATE Notee SET " + materie + "= " +"'"+ readerValue + ", " + nota + "'" + " WHERE idElev =" + lastCell;

cmd.ExecuteNonQuery();

con.Close();

}

else

{

con.Close();

con.Open();

cmd.CommandText = "UPDATE Notee SET " + materie + "= " + "'" + nota + "'" + " WHERE idElev =" + lastCell;

cmd.ExecuteNonQuery();

con.Close();

}

}

con.Close();

noteeTableAdapter.Fill(noteDS1.Notee);

noteeBindingSource.Filter = "IdElev = " + lastCell;

A1(false);

}

private void btnRenuntare\_Click(object sender, EventArgs e)

{

A1(false);

}

private void btnMedie\_Click(object sender, EventArgs e)

{

A1(false);

DialogResult dR = MessageBox.Show("Doriti un raport despre situatia scoala al elevului?", "Medie",

MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dR == DialogResult.Yes)

{

RaportMedieForm f = new RaportMedieForm(this);

f.ShowDialog();

}

}

public string GetLastCell()

{

return lastCell;

}

private void A3()

{

con.ConnectionString = noteeTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Select DenumireClasa From Clase ";

con.Open();

r = cmd.ExecuteReader();

while (r.Read())

{

cbClasa.Items.Add(r.GetString(0));

}

con.Close();

}

}

}

**Medii**

**Graphical user interface

Description automatically generated**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 ProiectMPP

{

public partial class Medii : Form

{

private string lastCell;

OleDbConnection con = new OleDbConnection();

OleDbCommand cmd = new OleDbCommand();

OleDbDataReader r;

public Medii()

{

InitializeComponent();

}

private void Medii\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'noteDS.Notee' table. You can move, or remove it, as needed.

this.noteeTableAdapter.Fill(this.noteDS.Notee);

// TODO: This line of code loads data into the 'noteDS.Elev' table. You can move, or remove it, as needed.

this.elevTableAdapter.Fill(this.noteDS.Elev);

dataGridView1.Focus();

this.dataGridView1.Sort(this.dataGridView1.Columns["Nume"], ListSortDirection.Ascending);

if (dataGridView1.Rows.Count > 0)

{

lastCell = dataGridView1.Rows[0].Cells[0].Value.ToString();

noteeBindingSource.Filter = "IdElev = " + lastCell;

if(dataGridView2.Rows.Count > 0)

{

CalculMedie();

}

}

A3();

}

private void elevBindingSource\_CurrentChanged(object sender, EventArgs e)

{

noteeTableAdapter.Fill(noteDS.Notee);

if (dataGridView1.SelectedRows.Count > 0)

{

if (dataGridView1.SelectedRows != null && dataGridView1.SelectedRows[0].Cells[0].Value != null)

lastCell = dataGridView1.SelectedRows[0].Cells[0].Value.ToString();

noteeBindingSource.Filter = "IdElev = " + lastCell;

}

if(dataGridView2.Rows.Count > 0)

{

CalculMedie();

}

}

private void cbClasa\_ValueMemberChanged(object sender, EventArgs e)

{

string clasaAleasa = cbClasa.Text;

elevBindingSource.Filter = "Clasa = " + "'" + clasaAleasa + "'";

this.dataGridView1.Sort(this.dataGridView1.Columns["Nume"], ListSortDirection.Ascending);

dataGridView1.Focus();

if (dataGridView1.Rows.Count > 0)

{

lastCell = dataGridView1.Rows[0].Cells[0].Value.ToString();

noteeBindingSource.Filter = "IdElev = " + lastCell;

}

}

private void CalculMedie()

{

for (int i = 1; i < dataGridView2.Columns.Count - 1; i++)

{

string note = dataGridView2.Rows[0].Cells[i].Value.ToString();

int nrNote = 0;

int numar = 0;

bool eGol = false;

if (note == "")

{

eGol = true;

}

for (int j = 0; j < note.Length; j++)

{

if (Char.IsDigit(note[j]))

{

if (note[j] == '1' && note[j + 1] == '0')

{

char[] zece = { note[j], note[j + 1] };

string zecE = new string(zece);

numar += int.Parse(zecE);

nrNote++;

j++;

}

else

{

numar += int.Parse(note[j].ToString());

nrNote++;

}

}

}

if (eGol)

{

dataGridView2.Rows[0].Cells[i].Value = "";

}

else

{

dataGridView2.Rows[0].Cells[i].Value = numar / nrNote;

}

}

}

private void A3()

{

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Select DenumireClasa From Clase ";

con.Open();

r = cmd.ExecuteReader();

while (r.Read())

{

cbClasa.Items.Add(r.GetString(0));

}

con.Close();

}

}

}

**Exmatriculare**

**Graphical user interface, application

Description automatically generated**

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 ProiectMPP

{

public partial class lblData : Form

{

OleDbConnection con = new OleDbConnection();

OleDbCommand cmd = new OleDbCommand();

OleDbDataReader r;

public lblData()

{

InitializeComponent();

}

private void Exmatriculare\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'exmatriculariDS.Exmatriculari' table. You can move, or remove it, as needed.

this.exmatriculariTableAdapter.Fill(this.exmatriculariDS.Exmatriculari);

// TODO: This line of code loads data into the 'exmatriculariDS.Exmatriculari' table. You can move, or remove it, as needed.

this.exmatriculariTableAdapter.Fill(this.exmatriculariDS.Exmatriculari);

// TODO: This line of code loads data into the 'claseDS.Elev' table. You can move, or remove it, as needed.

this.elevTableAdapter.Fill(this.claseDS.Elev);

// TODO: This line of code loads data into the 'exmatriculariDS.Exmatriculari' table. You can move, or remove it, as needed.

this.exmatriculariTableAdapter.Fill(this.exmatriculariDS.Exmatriculari);

this.dataGridView2.Sort(this.dataGridView2

.Columns["Nume"], ListSortDirection.Ascending);

A3();

}

private void A1(bool v)

{

btnConfirmare.Enabled = v;

btnRenuntare.Enabled = v;

dateTimePicker1.Visible = v;

label3.Visible = v;

lblMotiv.Visible = v;

txtMotiv.Visible = v;

btnPromovare.Enabled = !v;

btnExmatriculare.Enabled = !v;

dataGridView1.Visible = !v;

}

private void A3()

{

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Select DenumireClasa From Clase ";

con.Open();

r = cmd.ExecuteReader();

while (r.Read())

{

cbClasa.Items.Add(r.GetString(0));

}

con.Close();

}

private void btnExmatriculare\_Click(object sender, EventArgs e)

{

A1(true);

lblMain.Text = "EXMATRICULARE";

txtMotiv.Enabled = true;

}

private void btnPromovare\_Click(object sender, EventArgs e)

{

A1(true);

lblMain.Text = "PROMOVARE";

txtMotiv.Text = "Promovare";

txtMotiv.Enabled = false;

}

private void cbClasa\_SelectedValueChanged(object sender, EventArgs e)

{

if (dataGridView2.Enabled)

{

string clasaAleasa = cbClasa.Text;

elevBindingSource.Filter = "Clasa = " + "'" + clasaAleasa + "'";

this.dataGridView2.Sort(this.dataGridView2.Columns["Nume"], ListSortDirection.Ascending);

dataGridView2.Focus();

}

}

private void btnRenuntare\_Click(object sender, EventArgs e)

{

A1(false);

lblMain.Text = "";

txtMotiv.Enabled = true;

}

private void btnConfirmare\_Click(object sender, EventArgs e)

{

if(lblMain.Text == "EXMATRICULARE")

{

if(txtMotiv.Text == "")

{

MessageBox.Show("Motivul nu a fost adaugat!", "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

int i = dataGridView2.CurrentCell.RowIndex;

string lastCell = dataGridView2.Rows[i].Cells[0].Value.ToString();

int lastCellInt = int.Parse(lastCell);

con.ConnectionString = exmatriculariTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

string listaCampuri = "IdElev, NumeComplet, Motiv, DataExm";

string listaValori =

lastCellInt + ", '" + dataGridView2.Rows[i].Cells[1].Value.ToString()

+ " " + dataGridView2.Rows[i].Cells[2].Value.ToString()

+ ". " + dataGridView2.Rows[i].Cells[3].Value.ToString() +"'" +

" ,'" + txtMotiv.Text + "'" +

" , '" + dateTimePicker1.Value.ToShortDateString() + "'";

cmd.CommandText = "Insert into Exmatriculari(" + listaCampuri + ") " +

"Values " + "(" + listaValori + ")";

MessageBox.Show(cmd.CommandText);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "DELETE FROM Elev Where IdElev= " + lastCellInt;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "DELETE FROM Notee Where IdElev= " + lastCellInt;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

elevTableAdapter.Fill(claseDS.Elev);

exmatriculariTableAdapter.Fill(exmatriculariDS.Exmatriculari);

}

else

{

int i = dataGridView2.CurrentCell.RowIndex;

string lastCell = dataGridView2.Rows[i].Cells[0].Value.ToString();

int lastCellInt = int.Parse(lastCell);

string clasa = dataGridView2.Rows[i].Cells[4].Value.ToString();

var clasaString = clasa;

var clasaStringCommands = clasaString.Split(new[] { ' ' }, 2);

var clasaNr = clasaStringCommands[0];

var clasaSimbol = clasaStringCommands[1];

string clasaNoua;

SituatieScoalara(lastCellInt);

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Select IdClasa From Clase Where DenumireClasa = " + "'" + clasa + "'";

con.Open();

int clasaId = (int)cmd.ExecuteScalar() + 3;

con.Close();

cmd.CommandText = "Update SituatieElev Set IdClasa = " + clasaId + " Where IdElev=" + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

con.Close();

cmd.CommandText = "DELETE FROM Notee Where IdElev= " + lastCellInt;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

if (clasaNr.ToString() == "IX")

{

clasaNoua = "X ";

clasaNoua += clasaSimbol.ToString();

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Update Elev Set Clasa = " + "'"+ clasaNoua + "'" + " Where IdElev=" + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "Update SituatieElev Set IdClasa = " + clasaId + " Where IdElev=" + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

}

else if(clasaNr.ToString() == "X")

{

clasaNoua = "XI ";

clasaNoua += clasaSimbol.ToString();

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Update Elev Set Clasa = " + "'" + clasaNoua + "'" + " Where IdElev=" + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "Update SituatieElev Set IdClasa = " + clasaId + " Where IdElev=" + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

}

else if(clasaNr.ToString() == "XI")

{

clasaNoua = "XII ";

clasaNoua += clasaSimbol.ToString();

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Update Elev Set Clasa = " + "'" + clasaNoua + "'" + " Where IdElev=" + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "Update SituatieElev Set IdClasa = " + clasaId + " Where IdElev=" + lastCellInt;

elevTableAdapter.Update(claseDS.Elev);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

}

else

{

con.ConnectionString = exmatriculariTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

string listaCampuri = "IdElev, NumeComplet, Motiv, DataExm";

string listaValori =

lastCellInt + ", '" + dataGridView2.Rows[i].Cells[1].Value.ToString()

+ " " + dataGridView2.Rows[i].Cells[2].Value.ToString()

+ ". " + dataGridView2.Rows[i].Cells[3].Value.ToString() + "'" +

" ,'" + txtMotiv.Text + "'" +

" , '" + dateTimePicker1.Value.ToShortDateString() + "'";

cmd.CommandText = "Insert into Exmatriculari(" + listaCampuri + ") " +

"Values " + "(" + listaValori + ")";

con.Open();

cmd.ExecuteNonQuery();

con.Close();

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "DELETE FROM Elev Where IdElev= " + lastCellInt;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "DELETE FROM Notee Where IdElev= " + lastCellInt;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

cmd.CommandText = "DELETE FROM SituatieElev Where IdElev= " + lastCellInt;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

elevTableAdapter.Fill(claseDS.Elev);

exmatriculariTableAdapter.Fill(exmatriculariDS.Exmatriculari);

}

}

elevTableAdapter.Update(claseDS.Elev);

elevTableAdapter.Fill(claseDS.Elev);

A1(false);

lblMain.Text = "";

txtMotiv.Enabled = true;

}

private void SituatieScoalara(int lastCell)

{

con.ConnectionString = elevTableAdapter.Connection.ConnectionString;

cmd.Connection = con;

cmd.CommandText = "Select \* from Notee Where IdElev = " + lastCell;

con.Open();

r = cmd.ExecuteReader();

r.Read();

for(int i = 1;i <= 11; i++)

{

if (r[i].ToString() == "")

{

MessageBox.Show("Elevul respectiv nu are situatia incheiata!"

, "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

if (r[i].ToString().Length < 4)

{

MessageBox.Show("Elevul respectiv nu are numarul de note suficente!"

, "Eroare", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

}

con.Close();

}

}

}