Colegiul Naţional “Tudor Vladimirescu”

Tg-Jiu

**PROIECT**

**DE ATESTARE A COMPETENTELOR PROFESIONALE LA INFORMATICÅ**

Profesor îndrumător: Elev:

Nodea Eugen Noroc Andrei-Mihail

Clasa: a XII-a A

~ 2019 ~

**EXAMENUL PENTRU CERTIFICAREA COMPETENŢELOR PROFESIONALE  
ÎN ÎNVĂŢĂMÂNTUL LICEAL**

**Anexa 1**

**Unitatea de învăţământ : *COLEGIUL NAŢIONAL „TUDOR VLADIMIRESCU” Tg- Jiu***

**Anul şcolar 2018-2019**

**REFERAT**

**ASUPRA PROIECTULUI REALIZAT DE ELEVUL (ELEVA)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**CLASA a XII-a \_\_**

Denumirea temei\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Nivelul documentării

|  |  |  |  |
| --- | --- | --- | --- |
| Insuficient | Satisfăcător | Bun | Foarte Bun |

1. Unitate

|  |  |  |
| --- | --- | --- |
| Soft educaţional | Utilitar | Altele |

1. Redactarea lucrării

|  |  |  |
| --- | --- | --- |
| Suficient | Bună | Foarte bună |

1. Partea grafică, desene

|  |  |  |
| --- | --- | --- |
| Suficient | Bună | Foarte bună |

1. Realizare practică
   * complexitate

|  |  |  |
| --- | --- | --- |
| Simplu | Mediu | Complex |

* + funcţionare

|  |  |
| --- | --- |
| DA | NU |

1. Bibliografie
   * cărţi de specialitate
   * manuale
   * reviste de specialitate

Concluzia : Se propune nota/calificativul : \_\_\_\_\_\_\_\_\_\_\_

**Profesor îndrumător,**

**Prof. \_\_\_\_\_\_\_\_\_\_\_\_**

**TEMA**

**ATM BANCOMAT**

**CUPRINS**

* Motivarea alegerii temei
* Descrierea aplicatiei
* Codul sursa al programului
* Concluzii
* Bibliografie
* Anexe

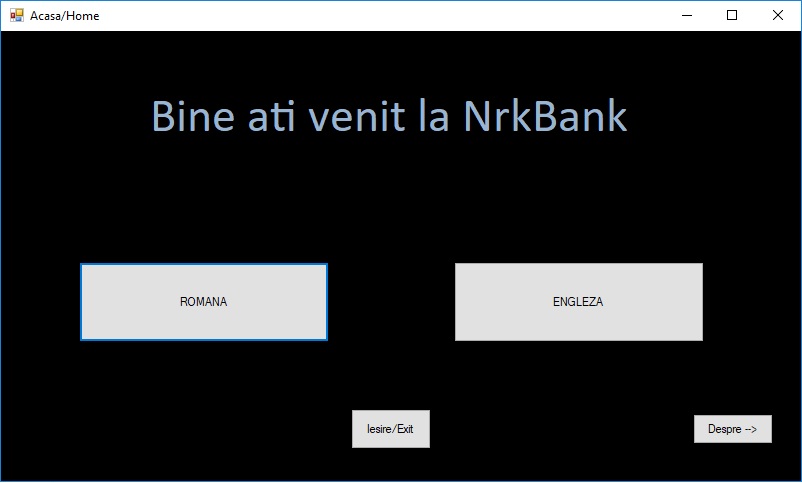
**Motivarea alegerii temei**

Am ales aceasta tema pentru o mai buna observare a modului in care bancomatele functioneaza. Aceste aparate electronice sunt folosite zilnic de catre oameni insa doar o mica parte din acestia si-au pus intreabarea sau stiu cum functioneaza.

In functie de dotarile acestora, ele pot efectua diferite operatiuni: retrageri, acceptari de depozite folosind bani lichizi sau cecuri, transferuri de fonduri, efectuarea platilor unor facturi, etc. Plasarea lor in diferite locuri faciliteaza accesarea veniturilor individului. In functie de banca careia apartin, acestea impun diferite comisioane pentru operatiunile desfasurate.

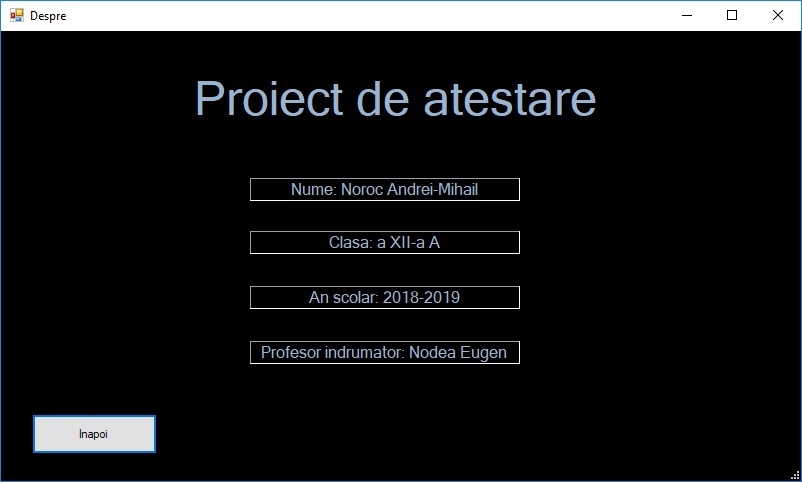
**Descrierea aplicatiei**

**Form-ul Acasa**

****

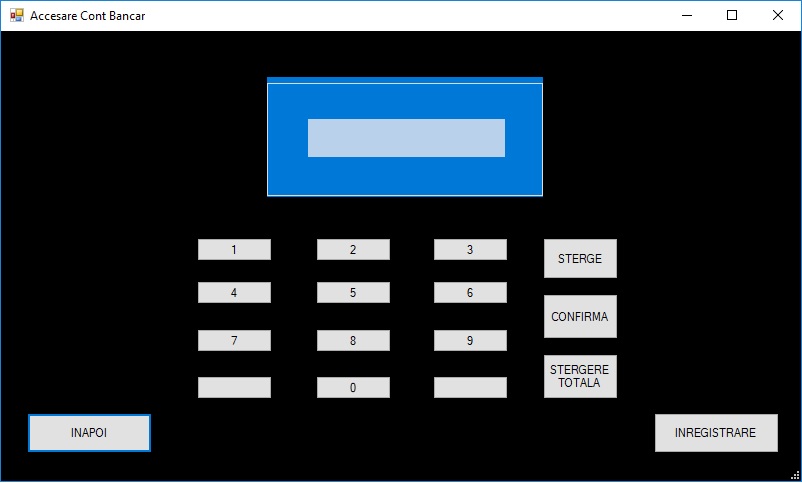
Windows Form-ul Acasa este cel cu care aplicatia porneste si prezinta cele doua variante pentru limba. De asemenea exista un buton pentru parasirea aplicatiei si altul care ne redirectioneaza catre un nou form unde putem gasi informatii in legatura cu elevul absolvent.

**Form-ul Despre**

****

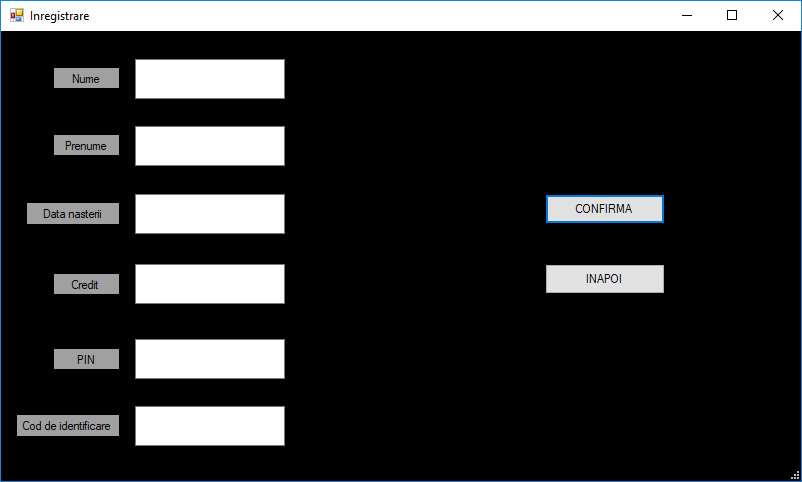
Form-ul Despre contine informatii in legatura cu autorul proiectului.

**Form-ul Accesare Cont Bancar (inainte de conectare)**

****

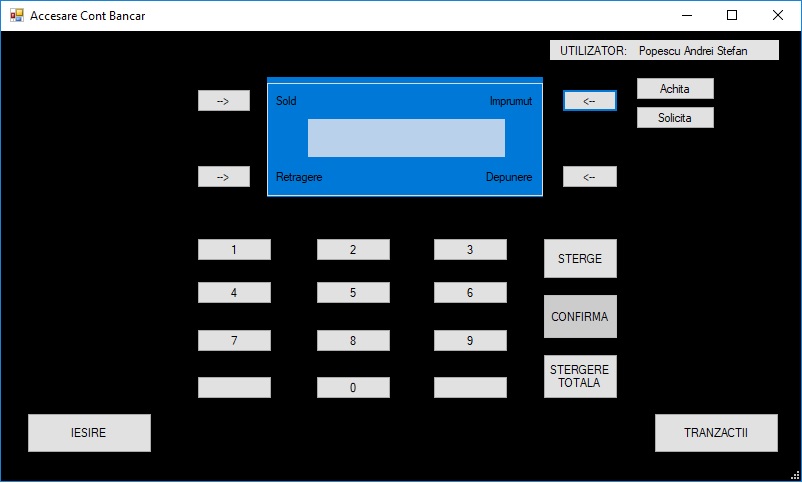
Form-ul (inainte de conectare) prezinta un ecran cu o tastatura specifica ATM-ului pentru a putea accesa contul bancar prin introducerea codului PIN. PIN-ul este verificat cu ajutorul interogarii de date SQL, afisandu-se un mesaj in functie de corectitudinea acestuia. In aceasta fereastra mai putem gasi un buton pentru inregistrare (creeare cont bancar) si altul pentru intoarcere la Form-ul Acasa.

**Form-ul Inregistrare**



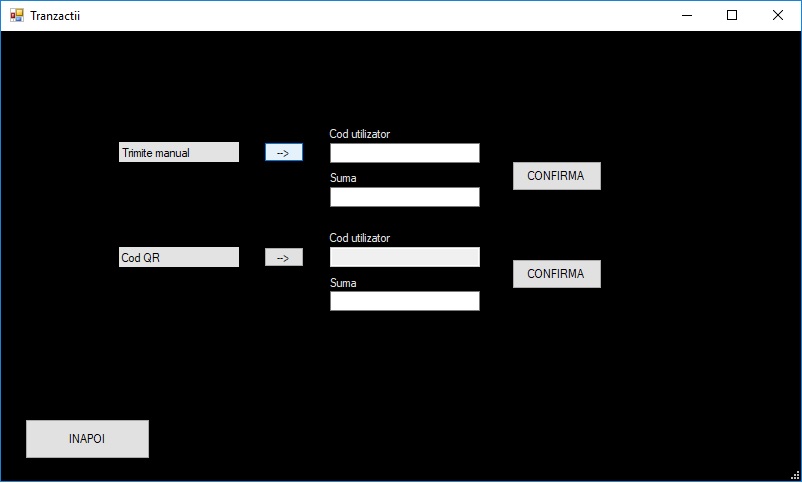
Form-ul Inregistrare este alcatuit din sase casete de text cu diferite proprietati carora li se alatura cate o eticheta, un buton pentru a ne intoarce la accesarea contului si un buton pentru a confirma operatiunea de inregistrare in baza de date a bancii care afiseaza un mesaj in functie de corectitudinea informatiilor din casetele de text.

**Form-ul Accesare Cont Bancar (dupa conectare)**

****

Form-ul Accesare Cont Bancar (dupa conectare) prezinta o eticheta unde este afisat numele utilizatorului, un ecran cu patru butoane pentru a efectua diferite operatiuni dupa ce a fost introdusa suma de bani, o tastatura cu butonul CONFIRMA blocat si doua butoane in colturile din josul ferestrei: unul pentru iesirea din aplicatie si unul pentru a efectua tranzactii.

**Tranzactii**

****

Form-ul Tranzactii doua modalitati de tranzactionare a unei sume de bani si un buton de intoarcere la Accesare Cont Bancar. Prima modalitate este prin introducerea codului de utilizator care este setat la infintarea contului si al doilea prin scanarea codului (ex: facturi).

**Codul sursa al programului**

**Acasa/Home Form**

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 ATMAPP

{

public partial class Form2 : Form

{

public Form2()

{

InitializeComponent();

}

private void Form2\_Load(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

Form3 f3 = new Form3();

f3.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

DialogResult iesire;

iesire = MessageBox.Show("Doriti sa parasiti aplicatia?", "ATM BANCA", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (iesire == DialogResult.Yes)

{

Application.Exit();

}

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void button4\_Click(object sender, EventArgs e)

{

Form5 f5 = new Form5();

f5.ShowDialog();

}

}

}

**Accesare Cont Bancar Form**

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;

namespace ATMAPP

{

public partial class Form3 : Form

{

SqlConnection con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\source\repos\ATMAPP\ATMAPP\Info.mdf;Integrated Security=True");

public Form3()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

Form2 f2 = new Form2();

f2.ShowDialog();

}

private void button4\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "1";

}

private void button5\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "2";

}

private void button6\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "3";

}

private void button7\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "4";

}

private void button8\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "5";

}

private void button9\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "6";

}

private void button10\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "7";

}

private void button11\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "8";

}

private void button12\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "9";

}

private void button14\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = CONFIRMA1.Text + "0";

}

private void button15\_Click(object sender, EventArgs e)

{

if (CONFIRMA1.Text.Length > 0)

CONFIRMA1.Text = CONFIRMA1.Text.Substring(0, CONFIRMA1.Text.Length - 1);

}

private void CONFIRMA1\_Click(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

this.Hide();

Form4 f4 = new Form4();

f4.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

SqlDataAdapter sda = new SqlDataAdapter("Select UserID, Nume, Prenume From UserInfo where PIN='" + CONFIRMA1.Text + "' ", con);

DataTable dt = new DataTable();

sda.Fill(dt);

if (dt.Rows.Count != 0)

{

MessageBox.Show("Autentificarea a fost realizata cu succes!");

label7.Text = dt.Rows[0][0].ToString();

label5.Text = dt.Rows[0][1].ToString() + " " + dt.Rows[0][2].ToString();

label5.Visible = true;

button19.Visible = true;

button20.Visible = true;

label1.Visible = true;

label2.Visible = true;

label4.Visible = true;

label3.Visible = true;

button21.Visible = true;

button22.Visible = true;

button2.Visible = false;

label6.Visible = true;

button3.Enabled = false;

button1.Enabled = false;

button1.Visible = false;

button24.Visible = true;

button24.Enabled = true;

button2.Enabled = false;

button25.Visible = true;

button25.Enabled = true;

}

else

{

MessageBox.Show("Cod pin gresit! Va rugam incercati din nou cu atentie!");

}

CONFIRMA1.Text = "";

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void label5\_Click(object sender, EventArgs e)

{

}

private void button19\_Click(object sender, EventArgs e)

{

SqlDataAdapter sda = new SqlDataAdapter("Select Credit, Datorii From UserInfo where UserID='" + label7.Text + "' ", con);

DataTable dt = new DataTable();

sda.Fill(dt);

MessageBox.Show("Sold curent:" + dt.Rows[0][0].ToString() + ", Datorii:" + dt.Rows[0][1].ToString());

}

private void button20\_Click(object sender, EventArgs e)

{

if (CONFIRMA1.Text != "")

{

DialogResult result = MessageBox.Show("Doriti sa efectuati operatia?", "Confirmation", MessageBoxButtons.YesNoCancel);

if (result == DialogResult.Yes)

{

SqlDataAdapter sda = new SqlDataAdapter("Select Credit From UserInfo where UserID='" + label7.Text + "' ", con);

DataTable dt = new DataTable();

sda.Fill(dt);

int sumafinala = StringToInt(CONFIRMA1.Text);

int cont = StringToInt(dt.Rows[0][0].ToString());

if (sumafinala > cont)

{

MessageBox.Show("Fonduri insuficiente!");

CONFIRMA1.Text = "";

}

else

{

cont -= sumafinala;

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "Update UserInfo Set Credit = @Cont Where UserID='" + label7.Text + "'";

cmd.Parameters.AddWithValue("@Cont", cont);

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Tranzactia a fost efectuata cu succes! Fonduri:" + cont);

CONFIRMA1.Text = "";

}

}

else if (result == DialogResult.No) CONFIRMA1.Text = "";

}

}

private void button17\_Click(object sender, EventArgs e)

{

CONFIRMA1.Text = "";

}

private void groupBox1\_Enter(object sender, EventArgs e)

{

}

private void button22\_Click(object sender, EventArgs e)

{

if (CONFIRMA1.Text != "")

{

DialogResult result = MessageBox.Show("Doriti sa efectuati operatia?", "Confirmation", MessageBoxButtons.YesNoCancel);

if (result == DialogResult.Yes)

{

SqlDataAdapter sda = new SqlDataAdapter("Select Credit From UserInfo where UserID='" + label7.Text + "' ", con);

DataTable dt = new DataTable();

sda.Fill(dt);

int sumafinala = StringToInt(CONFIRMA1.Text) + StringToInt(dt.Rows[0][0].ToString());

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "Update UserInfo Set Credit = @Cont Where UserID='" + label7.Text + "'";

cmd.Parameters.AddWithValue("@Cont", sumafinala);

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Tranzactia a fost efectuata cu succes! Fonduri:" + sumafinala);

CONFIRMA1.Text = "";

}

else if (result == DialogResult.No) CONFIRMA1.Text = "";

}

}

private void button21\_Click(object sender, EventArgs e)

{

if (button18.Visible == true && button23.Visible == true)

{

button18.Visible = false;

button23.Visible = false;

}

else

{

button18.Visible = true;

button23.Visible = true;

}

}

private void button23\_Click(object sender, EventArgs e)

{

if (CONFIRMA1.Text != "")

{

DialogResult result = MessageBox.Show("Doriti sa efectuati operatia?", "Confirmation", MessageBoxButtons.YesNoCancel);

if (result == DialogResult.Yes)

{

SqlDataAdapter sda = new SqlDataAdapter("Select Credit, Datorii From UserInfo where UserID='" + label7.Text + "' ", con);

DataTable dt = new DataTable();

sda.Fill(dt);

int datorii = StringToInt(dt.Rows[0][1].ToString());

int credit = StringToInt(dt.Rows[0][0].ToString());

int suma = StringToInt(CONFIRMA1.Text);

datorii += suma;

credit += suma;

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "Update UserInfo Set Datorii = @Datorie, Credit = @Cont Where UserID='" + label7.Text + "'";

cmd.Parameters.AddWithValue("@Datorie", datorii);

cmd.Parameters.AddWithValue("@Cont", credit);

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Tranzactia a fost efectuata cu succes! Datorii:" + datorii);

CONFIRMA1.Text = "";

}

else if (result == DialogResult.No) CONFIRMA1.Text = "";

}

}

int StringToInt(string SIR)

{

int k = SIR.Length - 1;

int p = 1;

int rez1 = 0;

while (k >= 0 && SIR[k] == '0')

{

p = p \* 10;

--k;

}

for (; k >= 0; --k) rez1 = rez1 \* 10 + SIR[k] - '0';

int rezultat = 0;

while (rez1 > 0)

{

rezultat = rezultat \* 10 + (rez1 % 10);

rez1 /= 10;

}

rezultat \*= p;

return rezultat;

}

private void button18\_Click(object sender, EventArgs e)

{

if (CONFIRMA1.Text != "")

{

DialogResult result = MessageBox.Show("Doriti sa efectuati operatia?", "Confirmation", MessageBoxButtons.YesNoCancel);

if (result == DialogResult.Yes)

{

SqlDataAdapter sda = new SqlDataAdapter("Select Credit, Datorii From UserInfo where UserID='" + label7.Text + "' ", con);

DataTable dt = new DataTable();

sda.Fill(dt);

int credit = StringToInt(dt.Rows[0][0].ToString());

int suma = StringToInt(CONFIRMA1.Text);

int datorii = StringToInt(dt.Rows[0][1].ToString());

if (datorii == 0) MessageBox.Show("Nu exista datorii!");

else

{

if (suma <= credit)

{

if (suma >= datorii)

{

credit -= datorii;

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "Update UserInfo Set Datorii = @Datorie, Credit = @Cont Where UserID='" + label7.Text + "'";

cmd.Parameters.AddWithValue("@Datorie", 0);

cmd.Parameters.AddWithValue("@Cont", credit);

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Tranzactia a fost efectuata cu succes! Datorii:" + 0);

}

else

{

credit -= suma;

datorii -= suma;

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "Update UserInfo Set Datorii = @Datorie, Credit = @Cont Where UserID='" + label7.Text + "'";

cmd.Parameters.AddWithValue("@Datorie", datorii);

cmd.Parameters.AddWithValue("@Cont", credit);

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Tranzactia a fost efectuata cu succes! Datorii:" + datorii);

}

}

else MessageBox.Show("Fonduri insuficiente!");

}

CONFIRMA1.Text = "";

}

else if(result == DialogResult.No) CONFIRMA1.Text = "";

}

}

private void button24\_Click(object sender, EventArgs e)

{

DialogResult iesire;

iesire = MessageBox.Show("Doriti sa parasiti aplicatia?", "ATM BANCA", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (iesire == DialogResult.Yes)

{

Application.Exit();

}

}

private void button25\_Click(object sender, EventArgs e)

{

Form1 form1 = new Form1(label7.Text);

form1.ShowDialog();

}

}

}

**Inregistrare Form**

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;

namespace ATMAPP

{

public partial class Form4 : Form

{

string connectionString = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\source\repos\ATMAPP\ATMAPP\Info.mdf;Integrated Security=True;";

public Form4()

{

InitializeComponent();

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox1.Text != "" && textBox2.Text != "" && textBox3.Text != "" && textBox4.Text != "" && textBox5.Text != "" && textBox6.Text != "")

{

SqlDataAdapter sda = new SqlDataAdapter("Select Credit From UserInfo where PIN='" + textBox5.Text + "' ", connectionString);

DataTable dt = new DataTable();

sda.Fill(dt);

if (dt.Rows.Count == 0)

{

using (SqlConnection sqlCon = new SqlConnection(connectionString))

{

sqlCon.Open();

SqlCommand sqlCmd = new SqlCommand("UserAdd", sqlCon);

sqlCmd.CommandType = CommandType.StoredProcedure;

sqlCmd.Parameters.AddWithValue("@Nume", textBox1.Text.Trim());

sqlCmd.Parameters.AddWithValue("@Prenume", textBox2.Text.Trim());

sqlCmd.Parameters.AddWithValue("@Dnastere", textBox3.Text.Trim());

sqlCmd.Parameters.AddWithValue("@Credit", textBox4.Text.Trim());

sqlCmd.Parameters.AddWithValue("@PIN", textBox5.Text.Trim());

sqlCmd.Parameters.AddWithValue("@Datorii", 0);

sqlCmd.Parameters.AddWithValue("@CodID", textBox6.Text.Trim());

sqlCmd.ExecuteNonQuery();

MessageBox.Show("Inregistrare realizata cu succes!");

textBox1.Text = textBox2.Text = textBox3.Text = textBox4.Text = textBox5.Text = textBox6.Text = "";

}

}

else

{

textBox5.Text = "";

MessageBox.Show("PIN-ul a fost deja utilizat!");

}

}

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

this.Hide();

Form3 f3 = new Form3();

f3.ShowDialog();

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

private void textBox4\_TextChanged(object sender, EventArgs e)

{

}

private void textBox5\_TextChanged(object sender, EventArgs e)

{

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsLetter(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

private void textBox2\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsLetter(e.KeyChar) && !char.IsControl(e.KeyChar) && !char.IsWhiteSpace(e.KeyChar))

{

e.Handled = true;

}

}

private void textBox3\_KeyPress(object sender, KeyPressEventArgs e)

{

}

private void textBox5\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

private void Form4\_Load(object sender, EventArgs e)

{

}

private void textBox4\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

private void textBox6\_TextChanged(object sender, EventArgs e)

{

}

private void textBox6\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

}

}

**Tranzactii Form**

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;

namespace ATMAPP

{

public partial class Form1 : Form

{

SqlConnection con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\source\repos\ATMAPP\ATMAPP\Info.mdf;Integrated Security=True");

public Form1(string Str\_Value)

{

InitializeComponent();

label1.Text = Str\_Value;

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void button3\_Click(object sender, EventArgs e)

{

if (textBox2.Visible == false)

{

textBox2.Visible = true;

textBox4.Visible = true;

button5.Visible = true;

label5.Visible = true;

label7.Visible = true;

}

else

{

textBox2.Visible = false;

textBox4.Visible = false;

button5.Visible = false;

label5.Visible = false;

label7.Visible = false;

}

}

private void button2\_Click(object sender, EventArgs e)

{

if (textBox1.Visible == false)

{

textBox1.Visible = true;

textBox3.Visible = true;

button4.Visible = true;

label4.Visible = true;

label6.Visible = true;

}

else

{

textBox1.Visible = false;

textBox3.Visible = false;

button4.Visible = false;

label4.Visible = false;

label6.Visible = false;

}

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

private void textBox2\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

private void textBox3\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

private void textBox4\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))

{

e.Handled = true;

}

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void button4\_Click(object sender, EventArgs e)

{

if (textBox1.Text != "" && textBox3.Text != "")

{

DialogResult result = MessageBox.Show("Doriti sa efectuati operatia?", "Confirmation", MessageBoxButtons.YesNoCancel);

if (result == DialogResult.Yes)

{

SqlDataAdapter sda = new SqlDataAdapter("Select Credit From UserInfo where CodID='" + textBox1.Text + "' ", con);

DataTable dt = new DataTable();

sda.Fill(dt);

if (dt.Rows.Count != 0)

{

int suma = StringToInt(textBox3.Text);

int contut = StringToInt(dt.Rows[0][0].ToString());

SqlDataAdapter rda = new SqlDataAdapter("Select Credit From UserInfo where UserID='" + label1.Text + "' ", con);

DataTable dtmeu = new DataTable();

rda.Fill(dtmeu);

int contmeu = StringToInt(dtmeu.Rows[0][0].ToString());

if (suma > contmeu)

{

MessageBox.Show("Fonduri insuficiente!");

textBox1.Text = textBox3.Text = "";

}

else

{

contmeu -= suma;

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "Update UserInfo Set Credit = @Cont Where UserID='" + label1.Text + "'";

cmd.Parameters.AddWithValue("@Cont", contmeu);

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

contut += suma;

SqlCommand cmd1 = new SqlCommand();

cmd1.CommandText = "Update UserInfo Set Credit = @Contut Where CodID='" + textBox1.Text + "'";

cmd1.Parameters.AddWithValue("@Contut", contut);

cmd1.Connection = con;

con.Open();

cmd1.ExecuteNonQuery();

con.Close();

MessageBox.Show("Tranzactia a fost efectuata cu succes! Fonduri:" + contmeu);

textBox1.Text = textBox3.Text = "";

}

}

else

{

MessageBox.Show("Cod utilizator gresit!");

textBox1.Text = textBox3.Text = "";

}

}

else if (result == DialogResult.No) textBox1.Text = textBox3.Text = "";

}

}

int StringToInt(string SIR)

{

int k = SIR.Length - 1;

int p = 1;

int rez1 = 0;

while (k >= 0 && SIR[k] == '0')

{

p = p \* 10;

--k;

}

for (; k >= 0; --k) rez1 = rez1 \* 10 + SIR[k] - '0';

int rezultat = 0;

while (rez1 > 0)

{

rezultat = rezultat \* 10 + (rez1 % 10);

rez1 /= 10;

}

rezultat \*= p;

return rezultat;

}

private void label2\_Click(object sender, EventArgs e)

{

}

}

}

**Despre Form**

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 ATMAPP

{

public partial class Form5 : Form

{

public Form5()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void label1\_Click(object sender, EventArgs e)

{

}

}

}

**Info Baza de date**

**UserInfo Tabel**

CREATE TABLE [dbo].[UserInfo] (

[UserID] INT IDENTITY (1, 1) NOT NULL,

[Nume] NVARCHAR (50) NULL,

[Prenume] NVARCHAR (50) NULL,

[Dnastere] VARCHAR (50) NULL,

[PIN] INT NULL,

[Credit] INT NULL,

[Datorii] INT NULL,

[CodID] INT NULL,

CONSTRAINT [PK\_UserInfo] PRIMARY KEY CLUSTERED ([UserID] ASC)

);

**UserAdd Procedura stocata**

CREATE PROC UserAdd

@Nume varchar(50),

@Prenume varchar(50),

@Dnastere varchar(50),

@Credit int,

@PIN int,

@Datorii int,

@CodID int

AS

INSERT INTO UserInfo(Nume,Prenume,Dnastere,Credit,PIN,Datorii,CodID)

VALUES (@Nume,@Prenume,@Dnastere,@Credit,@PIN,@Datorii,@CodID)

**Concluzii**

Realizarea acestui atestat m-a ajutat in dezvoltarea aptitudinilor legate de C# si progamul in care a fost executat proiectul, Visual Studio 2017.

Am asimilat mai multe informatii despre limbajul C# si modul de functionare al bazelor de date utilizand resurse web precum tutorialele si site-uri destinate programarii.

Cele mai importante cunostiinte acumulate au fost despre modul in care aceste ATM-uri functioneaza, avantajele si dezavantajele acestora.

**Bibliografie**

**Resurse web**

* <https://www.youtube.com/?gl=RO&hl=ro>
* <https://stackoverflow.com/>
* <https://www.google.ro/?hl=ro>

**Anexe**

Alaturi de acest dosar este atasat DVD-ul ce contine aplicatia realizata.