

**ŠIAULIŲ VALSTYBINĖS KOLEGIJOS  
VERSLO IR TECHNOLOGIJŲ FAKULTETO  
INFORMATIKOS MOKSLŲ KATEDRA**

Mikas Montvilas IST22 grupės studentas

**C# PROJEKTINIS DARBAS**

Objektinio programavimo projektinis darbas

Darbo vadovas:  
Doc. Dr. D. Dervinis

**Šiauliai, 2024**

## **Egzamino projekto užduotis**

### **Asmeninių finansų apskaita**

**Tikslas:** sukurti programą C# programavimo kalbą asmeninių finansų tvarkymui

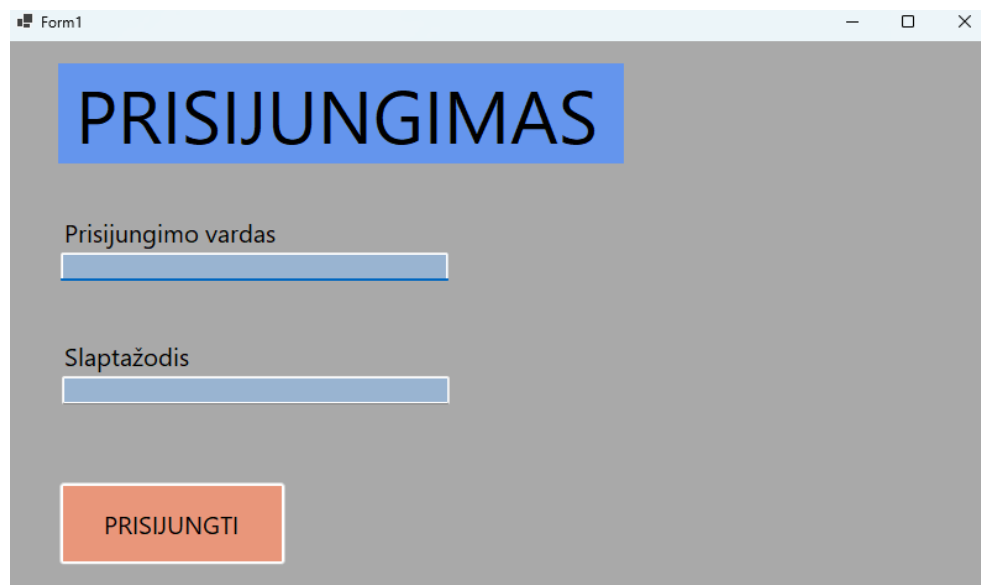
#### **Programos užduotys ir reikalavimai:**

1. Programa kuriama GUI Windows Forms arba WPF pagrindu
2. Duomenys saugomi MS SQL duomenų bazėje.
3. Programa paleidžiama atlikus autentifikavimą (galimas rinktis per DB arba nuosavą).
4. Turi būti galimybė įvesti pajamas ir išlaidas, nurodant kategoriją.
5. Įrašus turi būti galima pildyti, keisti, trinti.
6. Pajamų ir išlaidų kategorija pasirenkama iš klasifikatoriaus (dedikuota lentelė pvz su išlaidų įrašais: Maistas, Kuras; pajamos: atlyginimas, kita.). Klasifikatorių turi būti galimybė modifikuoti: pildyti, keisti, trinti.
7. Realus likučio, pajamų ir išlaidų suminis rodymas
8. Ataskaita bent pagal 3 pjūvius pvz.: periodo pasirinkimas, periodo ataskaita, pagal kategorijas suminė., gali būti pateikiama analizė (min, max, vidutinė kaina) ir pan.
9. Sudaryti projekto DB ir klasių UML diagramas.

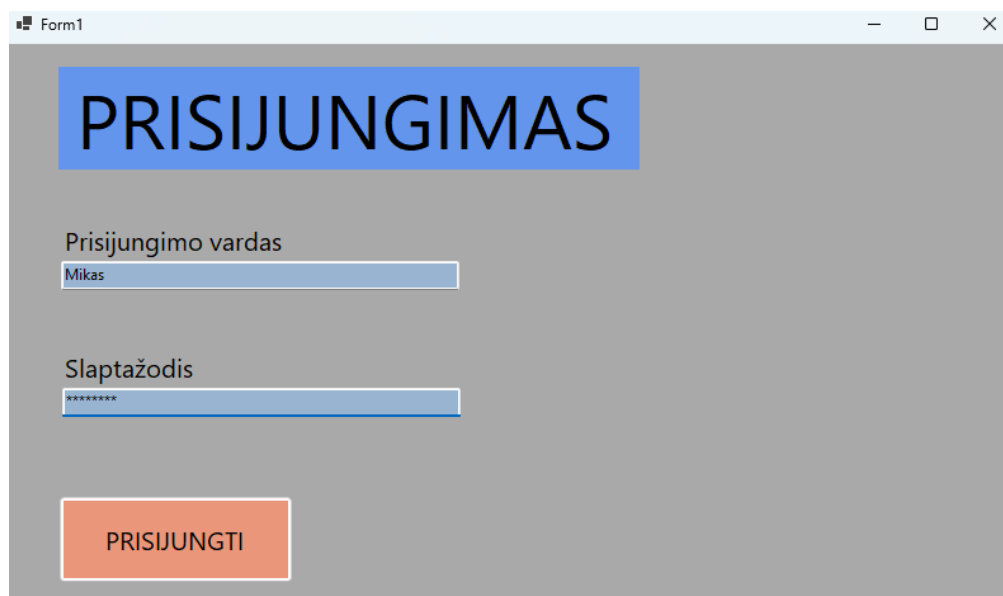
GitHub nuoroda: [https://github.com/MikasMont/Projektas\\_Objektinis](https://github.com/MikasMont/Projektas_Objektinis)

Programa buvo kuriama GUI Windows Forms C# pagrindu. Projekto užduotis buvo sukurti asmeninę pajamų ir išlaidų aplinką, kurioje yra: duomenys saugomi MS SQL duomenų bazėje; programa paleidžiama atlikus autentifikavimą (padarytas DB autentifikavimas ir sukurtas Form1 prisijungimas); turi būti galimybė įvesti pajamas ir išlaidas, nurodant kategoriją; įrašus turi būti galima pildyti, keisti, trinti; pajamų ir išlaidų kategorija pasirenkama iš klasifikatoriaus (dedikuota lentelė pvz.: su išlaidų įrašais: Maistas, Kuras; pajamos: atlyginimas, ir t.t); realaus likučio, pajamų ir išlaidų suminis rodymas ir ataskaita bent pagal 3 pjūvius pvz.: laikotarpio pasirinkimas, kategorijos pasirinkimas ir lėšų pasirinkimas (išlaidos, pajamos).

Paleidus kodą pirma naudotojas nukreipiamas į prisijungimo lauką. Naudotojas gali prisijungti tik tada kai jo prisijungimas yra duomenų bazėje. Įvedus tinkamus duomenis vartotojas nukeliamas į kitą formą (aplinką).

The image shows a Windows Forms application window titled 'Form1'. The background is a solid grey color. At the top, there is a blue rectangular box containing the word 'PRISIJUNGIMAS' in white, bold, sans-serif capital letters. Below this box, there are two text labels: 'Prisijungimo vardas' and 'Slaptažodis'. Each label is positioned to the left of a white rectangular input field with a thin blue border. Below the input fields, there is a red rectangular button with the text 'PRISIJUNGTI' in white, bold, sans-serif capital letters. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

1 pav. Prisijungimo laukas\_v1

This image is similar to the previous one, showing the same 'Form1' window with the 'PRISIJUNGIMAS' title. The 'Prisijungimo vardas' input field now contains the text 'Mikas'. The 'Slaptažodis' input field contains a series of asterisks '\*\*\*\*\*'. The 'PRISIJUNGTI' button remains in the same position at the bottom of the form. The window controls are still visible in the top right corner.

2 pav. Prisijungimo laukas\_v2

Prisijungimo formos kodas:

```
public partial class Form1 : Form
{
    // Prisijungimas į bazę
    private string connectionString = @"Data Source=localhost;Initial Catalog=Projektas;User ID=sa;Password=sa";

    1 reference
    public Form1()
    {
        InitializeComponent();
    }

    1 reference
    private void button1_Click(object sender, EventArgs e)
    {
        SqlConnection con = new SqlConnection(connectionString);
        {
            // Tikrinimas, pasižiūri ar jau yra toks vartotojas duomenų bazės lentelėje
            // Prisijungimai du: Mikas, mikas123 arba Jonas, jonas123
            string query = "SELECT COUNT(1) FROM Prisijungimai WHERE Vardas=@Vardas AND Slaptažodis=@Slaptažodis";
            SqlCommand cmd = new SqlCommand(query, con);
            cmd.Parameters.AddWithValue("@Vardas", textBox2.Text);
            cmd.Parameters.AddWithValue("@Slaptažodis", textBox1.Text);
            con.Open();
            int count = Convert.ToInt32(cmd.ExecuteScalar());
            if (count == 1)
            {
                // Paslėpti prisijungimo formą
                this.Hide();
                // Rodyti pagrindinę formą
                Form2 mainForm = new Form2();
                mainForm.Show();
            }
            else
            {
                MessageBox.Show("Įvestas neteisingas naudotojo vardas arba slaptažodis.", "Ispėjimas", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
            }
        }
    }
}
```

3 pav. Form1 pradžios/prisijungimo kodas

```
1 reference
private void textBox1_TextChanged(object sender, EventArgs e)
{
    // Pakeičia simbolius į žvaigždutes
    textBox1.PasswordChar = '*';
}
```

4 pav. Slaptažodžio lauko kodas

Čia yra pirmos formos kodas, kuriame yra aprašytas prisijungimas į SQL, kuriame paskui žiūrima ar įvestas vardas ir slaptažodis yra sukurtose duomenų bazėje, jeigu taip jis nukreipiamas į kitą aplinką, jeigu ne yra išmetamas pranešimas, jog įvestas neteisingas naudotojo vardas arba slaptažodis, taipogi šiame kode yra naudojamas ir PasswordChar tai leidžia pakeisti slaptažodžio lauko simbolius į bet kokius pasirinktus, jog pašaliniai praeiviai nematyti įrašomo slaptažodžio atvira forma.

Antroji ir pagrindinė forma atrodo taip:

Šioje aplinkoje galima daryti viską kas nurodytą uždaviniuose: galima trinti, keisti įrašą, matyti įrašytą įrašą, galimas pasirinkimas tarp kategorijų, šaltinių ir lėšų, galima įrašyti ir pastabas, matomas dabartinis sąskaitos, pajamų ir išlaidų likutis, taipogi galima gauti ataskaitas, kurios spausdinamos į PDF.

The screenshot shows the 'Form2' application window. On the left, there are input fields for 'Data' (set to '2024 m. gegužė 29 d., trečiadienis'), 'Suma', a radio button group for 'Pajamos' and 'Išlaidos', a 'Kategorija' dropdown, a 'Šaltinis' dropdown (set to 'Asmeninė sąskaita'), and a 'Pastaba' text area. On the right, there's a large grey rectangular area representing a data grid. Below this area are buttons for 'Rodyti duomenų bazę', 'KEISTI ĮRAŠĄ', and 'TRINTI ĮRAŠĄ'. Further down, summary statistics are displayed: 'Sąskaitos likutis - label10', 'Pajamos - label11', and 'Išlaidos - label12'. A 'PDF kūrimas' button is also present. At the bottom left is a 'PATEIKTI' button, and at the bottom right is a 'UŽDARYTI' button.

5 pav. Form2 aplinka

Rodoma informacija aplinkoje:

Šiame DataGridView lange pasirinkau rodyti tik vartotojui naudingą informaciją, nes jam tikrai nereikia matyti absoliučiai visų lentelių ID ar kokios perteklinės informacijos, čia taipogi rodoma informacija 2 savaitių intervale, jog neužkrauti programos.

This screenshot shows the same 'Form2' application window, but the data grid is populated with transaction data. The grid has columns: ID, Data, Suma, Pastaba, Pavadinimas, Šaltinis, and Tipo\_pav. The data is filtered to show transactions from the last two weeks. Below the grid, the summary statistics are updated: 'Sąskaitos likutis - 1978,5000', 'Pajamos - 4930,0000', and 'Išlaidos - -2951,5000'. The buttons and layout remain the same as in the previous screenshot.

ID	Data	Suma	Pastaba	Pavadinimas	Šaltinis	Tipo_pav
10	2024-05-24 08:31	1500,0000	Uždarbą	Atlyginimas	Asmeninė sąsk...	Pajamos
11	2024-05-24 11:26	-100,0000	Pilnas	Pramogos	Asmeninė sąsk...	Išlaidos
12	2024-05-24 08:31	-150,0000	Maistas savaitei	Maistas	Grinieji pinigai	Išlaidos
14	2024-05-17 11:30	-150,0000	Dažasvaidis	Pramogos	Grinieji pinigai	Išlaidos
15	2024-05-17 11:30	15,0000	Rasta gatveje	Kita	Grinieji pinigai	Pajamos
16	2024-05-27 09:15	400,0000	Premija	Kita	Asmeninė sąsk...	Pajamos
17	2024-05-27 09:27	-150,0000	Savaitei	Maistas	Grinieji pinigai	Išlaidos

6 pav. Form2 aplinka su matoma duomenų baze

Duomenų įrašymas paprastas, tiesiog užtenka įrašyti visus norimus duomenis į tinkamus laukus ir spausti pateikti, viskam įvykus sėkmingai išmetama lentelė pranešianti, jog viskas įkelta sėkmingai.

Form2

Data

2024 m. gegužė 29 d., trečiadienis

Suma

1500

Kategorija

Atlyginimas

Šaltinis

Asmeninė sąskaita

Pastaba

Gauta alga už gegužės mėn.

Duomenų bazėje duomenys yra rodomi dviejų savaitių intervale.

	ID	Data	Suma	Pastaba	Pavadinimas	Šaltinis	Tipo_pav
▶	10	2024-05-24 08:31	1500,0000	Uždarba	Atlyginimas	Asmeninė sąsk...	Pajamos
	11	2024-05-24 11:26	-100,0000	Pilnas	Pramogos	Asmeninė sąsk...	Išlaidos
	12	2024-05-24 08:31	-150,0000	Maistas savaitei	Maistas	Grininėji pinigai	Išlaidos
	14	2024-05-17 11:30	-150,0000	Dažsavaidis	Pramogos	Grininėji pinigai	Išlaidos
	15	2024-05-17 11:30	15,0000	Rasta gatvėje	Kita	Grininėji pinigai	Pajamos
	16	2024-05-27 09:15	400,0000	Premija	Kita	Asmeninė sąsk...	Pajamos
	17	2024-05-27 09:27	-150,0000	Savaitei	Maistas	Grininėji pinigai	Išlaidos

Rodyti duomenų bazę

KEISTI ĮRAŠĄ

TRINTI ĮRAŠĄ

Sąskaitos likutis - 1978,5000

Pajamos - 4930,0000

Išlaidos - -2951,5000

PDF kūrimas

Patvirtinimas

i

Viskas įkelta sėkmingai

Gera

PATEIKTI

UŽDARYTI

24

2024-05-29 12:57

1500,0000

Gauta alga už g...

Atlyginimas

Asmeninė sąsk...

Pajamos

7 pav. Form2 aplinka, informacijos pateikimas

Vedant duomenis realiu laiku taipogi pasikeičia ir sąskaitos likutis, rodamos pajamos ir išlaidos.

**Saskaitos likutis -** 3478,5000

**Pajamos -** 6430,0000

**Išlaidos -** -2951,5000

8 pav. Sąskaitos likučio laukas

Norint uždaryti programą užtenka paskausti uždaryti mygtuką ir patvirtinti uždarymą ir programa uždarys.

Form2

Antros formos kodas:

Antros formos kode taipogi yra prisijungimo į SQL duomenų bazę string. Čia taipogi yra ir inicijuoti ir elementai ir laikmatis. Tas laikmatis yra nustatytas dėl realaus likučio rodymo, jo intervalas yra 10 sekundžių tai vadinasi, jog sąskaitos likutis vis atnaujinamas kas dešimt sekundžių.

```
1 using System;
2 using System.Data.SqlClient;
3 using System.Data;
4 using System.Windows.Forms;
5 using static System.Windows.Forms.VisualStyles.VisualStyleElement;
6
7
8 namespace Projektas
9 {
10     4 references
11     public partial class Form2 : Form
12     {
13         // Prisijungimas į Microsoft SQL ir į duomenų
14         public SqlConnection cnn;
15         string connectionString = @"Data Source=localhost;Initial Catalog=Projektas;User ID=sa;Password=sa";
16         private System.Windows.Forms.Timer timer;
17         1 reference
18         public Form2()
19         {
20             InitializeComponent();
21             InitializeTimer();
22         }
23         1 reference
24         private void InitializeTimer()
25         {
26             timer = new System.Windows.Forms.Timer();
27             timer.Interval = 10000;
28             timer.Tick += new EventHandler(Keitimassum);
29             timer.Start();
30         }
31     }
32 }
```

10 pav. Form2 kodas

Toliau čia pats likučio pasikeitimo principas, keičiami yra label pavadinimai ir jie pasiimami iš duomenų bazės, čia yra sukurtas query, kuris pasiima jau sudėtus skaičius iš duomenų bazės, aiškų, yra pasiimami trys: visi pinigai, pajamos kur skaičiai didesni už nuli ir išlaidos kur skaičiai mažesni už nuli.

```
30 private void Keitimassum(object sender, EventArgs e)
31 {
32     using (SqlConnection connection = new SqlConnection(connectionString))
33     {
34         connection.Open();
35
36         // Update label10 sąskaitos likutis
37         string sumQuery = "SELECT SUM(Suma) FROM Pinigai";
38         using (SqlCommand sumCmd = new SqlCommand(sumQuery, connection))
39         {
40             label10.Text = sumCmd.ExecuteScalar()?.ToString() ?? "0";
41         }
42
43         // Update label11 pajamos
44         string positiveSumQuery = "SELECT SUM(Suma) FROM Pinigai WHERE Suma > 0";
45         using (SqlCommand positiveSumCmd = new SqlCommand(positiveSumQuery, connection))
46         {
47             label11.Text = positiveSumCmd.ExecuteScalar()?.ToString() ?? "0";
48         }
49
50         // Update label12 išlaidos
51         string negativeSumQuery = "SELECT SUM(Suma) FROM Pinigai WHERE Suma < 0";
52         using (SqlCommand negativeSumCmd = new SqlCommand(negativeSumQuery, connection))
53         {
54             label12.Text = negativeSumCmd.ExecuteScalar()?.ToString() ?? "0";
55         }
56     }
57 }
```

11 pav. Form2 kodas, įrašo keitimas



Toliau ši kodo dalis yra duomenų bazės rodymas ir įrašų trynimas. Duomenų bazės rodymui reikia turėti prisijungimą į duomenų bazę ir reikia aprašyti duomenis, kuriuos norime matyti, viskas daroma per dataGridView, jame yra rodoma duomenų bazės lentelė ir prieš tai aprašyta sąlyga, kuri rodo vartotojui specifinius duomenis. Norint trinti įrašą reikia pasirinkti visą įrašą ir pagal pasirinktą įrašą yra matomas įrašo ID ir pagal jį tada tas įrašas su visais duomenimis yra ištrynimas iš dataGridView ir iš duomenų bazės lentelės.

```

60 2 references
61 private void Rodymas_Click_1(object sender, EventArgs e)
62 {
63     // Rodyti duomenų bazę
64     SqlConnection cnn = new SqlConnection(connectionString);
65     cnn.Open();
66     string Query = "SELECT Pinigai.ID, Data, Suma, Pastaba, Pavadinimas, Saltinis, Tipo_pav from Pinigai inner join Klasifikatorius on Pinigai.Klasif_ID=K
67     SqlCommand cmd = new SqlCommand(Query, cnn);
68     var reader = cmd.ExecuteReader();
69     DataTable table = new DataTable();
70     table.Load(reader);
71     dataGridView1.DataSource = table;
72     cnn.Close();
73 }
74 1 reference
75 private void Trinimas_Click_1(object sender, EventArgs e)
76 {
77     //Trinti irasa (pasirenkant visą eilutę per DataGridView1)
78     if (dataGridView1.SelectedRows.Count == 0)
79         return;
80
81     string sql = "DELETE FROM Pinigai WHERE ID = @rowID";
82     SqlConnection cnn = new SqlConnection(connectionString);
83     SqlCommand deleteRecord = new SqlCommand(sql, cnn);
84     {
85         cnn.Open();
86
87         int selectedIndex = dataGridView1.SelectedRows[0].Index;
88         int rowID = Convert.ToInt32(dataGridView1[0, selectedIndex].Value);
89
90         deleteRecord.Parameters.Add("@rowID", SqlDbType.Int).Value = rowID;
91         deleteRecord.ExecuteNonQuery();
92
93         dataGridView1.Rows.RemoveAt(selectedIndex);

```

12 pav. Form2 kodas, duomenų bazės lentelės rodymas, įrašo trynimas

Ši kodo dalis aprašo įrašo keitimą, joje pasirinkus įrašą per dataGridView galima įrašą keisti. Principas paprastas reikia turėti prisijungimą į duomenų bazę ir naudoti komandą UPDATE, kuri pakeičia duomenų bazės lentelėje esančius duomenis, jie pasikeičia ir per dataGridView. Šiame kode yra išrašyti visi laukai, kurie gali būti keičiami.

```

97 private void Keitimas_Click(object sender, EventArgs e)
98 {
99     // Keisti įrašą (atnaujinti pasirinktą įrašą)
100     // Keičiant įrašą reikia pasirinkti visą eilutę per DataGridView ir surašyti norimus naujinimus į langelius ir spausti KEISTI ĮRAŠĄ.
101     if (dataGridView1.SelectedRows.Count == 0)
102         return;
103
104     DataGridViewRow selectedRow = dataGridView1.SelectedRows[0];
105     int rowID = Convert.ToInt32(selectedRow.Cells["ID"].Value);
106
107     string query = "UPDATE Pinigai SET Data = @Data, Suma = @Suma, Pastaba = @Pastaba, Klasif_ID = @Klasif_ID, Saltinis_ID = @Saltinis_ID WHERE ID = @ID";
108
109     String suma = sumos.Text;
110     int sumaskai;
111     int.TryParse(suma, out sumaskai);
112     DataRowView selectedRows = listBox1.SelectedItem as DataRowView;
113     int daugiklis = Convert.ToInt32(selectedRows["Daugiklis"]);
114     sumaskai *= daugiklis;
115
116     SqlConnection connection = new SqlConnection(connectionString);
117     {
118         SqlCommand cmd = new SqlCommand(query, connection);
119         cmd.Parameters.AddWithValue("@Data", dateTimePicker1.Value);
120         cmd.Parameters.AddWithValue("@Suma", sumaskai);
121         cmd.Parameters.AddWithValue("@Pastaba", textBox1.Text);
122         cmd.Parameters.AddWithValue("@Klasif_ID", (int)comboBox1.SelectedValue);
123         cmd.Parameters.AddWithValue("@Saltinis_ID", (int)comboBox2.SelectedValue);
124         cmd.Parameters.AddWithValue("@ID", rowID);
125
126         connection.Open();
127         cmd.ExecuteNonQuery();
128     }
129
130     // Atnaujinti DataGridView1
131     rodymas_Click_1(sender, e);
132     MessageBox.Show("Įrašas sėkmingai atnaujintas.", "Patvirtinimas", MessageBoxButtons.OK, MessageBoxIcon.Information);
133 }
134

```

13 pav. Form2 kodas, įrašo keitimas

Kita kodo dalis yra programos uždarymo ir duomenų pateikimo į duomenų bazės lentelę. Uždarymo kodas tiesiog yra naudojamas su button ir paspaudus šį mygtuką vartotojui duodamas pasirinkimas pilnai uždaryti programą arba grįžti atgal. Programoje taipogi veikia ir pateikimas, jis veikia taip: prisijungiama prie duomenų bazės tada vartotojui suvedus visus norimus duomenis, kode vyksta tikrinimas ar viskas yra parinkta teisingai ar nėra jokių klaidų, tada yra vykdomas query, kuris įkelia duomenis į duomenų bazės lentelę, ir jeigu viskas įkelta sėkmingai yra parodomas pranešimas, jog viskas įkelta.

```

137 private void uždarymas_Click_1(object sender, EventArgs e)
138 {
139     // Programos uždarymas (Form1 ir Form2)
140     DialogResult ats;
141     ats = MessageBox.Show("Ar norite uždaryti programą?", "Uždarymas", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
142     if (ats == DialogResult.Yes)
143     {
144         this.Close();
145         Environment.Exit(1);
146     }
147 }
148
149 1 reference
150 private void pateikimas_Click(object sender, EventArgs e)
151 {
152     if (comboBox1.SelectedValue == null || comboBox2.SelectedValue == null || listBox1.SelectedValue == null)
153     {
154         MessageBox.Show("Prašau pasirinkti tinkamą šaltinį ir kategoriją", "Klaida", MessageBoxButtons.OK, MessageBoxIcon.Error);
155         return;
156     }
157
158     SqlConnection connection = new SqlConnection(connectionString);
159     {
160         connection.Open();
161
162         DataRowView selectedRow = listBox1.SelectedItem as DataRowView;
163         if (selectedRow != null)
164         {
165             int daugiklis = Convert.ToInt32(selectedRow["Daugiklis"]);
166
167             string query = "INSERT INTO Pinigai (Data, Suma, Pastaba, Klasif_ID, Saltinis_ID) " +
168                 "VALUES (@Data, @Suma, @Pastaba, @Klasif_ID, @Saltinis_ID)";
169
170             SqlCommand cmd = new SqlCommand(query, connection);
171             {
172                 // Pakeisti suma į kitą formatą/value
173                 if (!decimal.TryParse(sumos.Text, out decimal suma))
174                 {
175                     MessageBox.Show("Netinkamas formatas.", "Klaida", MessageBoxButtons.OK, MessageBoxIcon.Error);
176                     return;
177                 }
178
179                 suma *= daugiklis;
180
181                 // Parametrai
182                 cmd.Parameters.AddWithValue("@Data", dateTimePicker1.Value);
183                 cmd.Parameters.AddWithValue("@Suma", suma);
184                 cmd.Parameters.AddWithValue("@Pastaba", textBox1.Text);
185                 cmd.Parameters.AddWithValue("@Klasif_ID", (int)comboBox1.SelectedValue);
186                 cmd.Parameters.AddWithValue("@Saltinis_ID", (int)comboBox2.SelectedValue);
187
188                 // Padaryti komandą
189                 try
190                 {
191                     cmd.ExecuteNonQuery();
192                     MessageBox.Show("Viskas įkelta sėkmingai", "Patvirtinimas", MessageBoxButtons.OK, MessageBoxIcon.Information);
193                 }
194                 catch (SqlException ex)
195                 {
196                     MessageBox.Show("Nepavyko įkelti įrašus: " + ex.Message, "Klaida", MessageBoxButtons.OK, MessageBoxIcon.Error);
197                 }
198             }
199         }
200     }
201 }

```

14 pav. Form2 kodas, programos uždarymas ir duomenų pateikimas

Form2\_Load vyksta duomenų įkėlimas į elementus iš duomenų bazės t.y. kai forma (aplinka) yra užkraunama, į aprašytus elementus yra įkeliami duomenys iš duomenų bazės, tai yra daroma, nes jog nereiktu kas kart keisti kodo atsiradus naujam pasirinkimui ar apibūdinimui. Čia yra naudojamos SELECT komandos, kurios pasiima reikiamus duomenis ir paskui juos įkelia į formos elementus.

```
215 private void Form2_Load(object sender, EventArgs e)
216 {
217
218     SqlConnection connection = new SqlConnection(connectionString);
219     {
220         connection.Open();
221
222         // Kategorija ComboBox (comboBox1)
223         string queryKategorija = "SELECT ID, Pavadinimas FROM Klasifikatorius";
224         SqlCommand cmd = new SqlCommand(queryKategorija, connection);
225         SqlDataReader reader = cmd.ExecuteReader();
226
227         DataTable kategorijaTable = new DataTable();
228         kategorijaTable.Load(reader);
229         comboBox1.DataSource = kategorijaTable;
230         comboBox1.DisplayMember = "Pavadinimas";
231         comboBox1.ValueMember = "ID";
232
233
234         // Saltinis ComboBox (comboBox2)
235         string querySaltinis = "SELECT ID, Saltinis FROM Saltinis";
236         SqlCommand cmda = new SqlCommand(querySaltinis, connection);
237         SqlDataReader readera = cmda.ExecuteReader();
238
239         DataTable saltinisTable = new DataTable();
240         saltinisTable.Load(readera);
241         comboBox2.DataSource = saltinisTable;
242         comboBox2.DisplayMember = "Saltinis";
243         comboBox2.ValueMember = "ID";
244
245
246
247         // Tipas ListBox1 (listBox1)
248         string queryTipas = "SELECT ID, Tipo_pav, Daugiklis FROM Tipas";
249         SqlCommand cmdas = new SqlCommand(queryTipas, connection);
250         SqlDataReader readeras = cmdas.ExecuteReader();
251
252         DataTable TipasTable = new DataTable();
253         TipasTable.Load(readeras);
254         listBox1.DataSource = TipasTable;
255         listBox1.DisplayMember = "Tipo_pav";
256         listBox1.ValueMember = "ID";
257
258         listBox1.SelectedIndexChanged += listBox1_SelectedIndexChanged;
259         comboBox1.DataSource = null;
260     }
```

15 pav. Form2 kodas, formos užkrovimas

Ši kodo dalis aprašo lėšų pasirinkimą, jog pasirinkus pajamas ar išlaidos keistųsi ir kategorijos, nes būtų nelogiška turėti algą prie išlaidų ir t.t. Čia naudojama taipogi SELECT užklausa, kuri pasiima reikiamus duomenis, bet prieš tai yra IF sąlyga pagal kurį pasirinktą variantą bus rodomi pasirinkimai.

```

265 private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
266 {
267     if (listBox1.SelectedValue != null)
268     {
269         DataRowView selectedRow = listBox1.SelectedItem as DataRowView;
270         if (selectedRow != null)
271         {
272             int selectedTipasId = Convert.ToInt32(selectedRow["ID"]);
273
274             SqlConnection connection = new SqlConnection(connectionString);
275             {
276                 connection.Open();
277
278                 // Query gauti geras kategorijas pagal pasirinktą tipą (pajamos ar išlaidos)
279                 string queryKategorija = "SELECT ID, Pavadinimas FROM Klasifikatorius WHERE Tipas_ID = @TipasID";
280                 SqlCommand cmd = new SqlCommand(queryKategorija, connection);
281                 {
282                     cmd.Parameters.AddWithValue("@TipasID", selectedTipasId);
283                     SqlDataReader reader = cmd.ExecuteReader();
284                     {
285                         DataTable kategorijaTable = new DataTable();
286                         kategorijaTable.Load(reader);
287                         comboBox1.DataSource = kategorijaTable;
288                         comboBox1.DisplayMember = "Pavadinimas";
289                         comboBox1.ValueMember = "ID";
290                     }
291                 }
292             }
293         }
294     }
295 }

```

16 pav. Form2 kodas, pasirinkimų keitimas

Ši kodo dalis priklauso keitimo programos daliai. Čia pagal pasirinktą įrašą į rašymo laukus yra įkeliami prieš tai parašyti duomenis, jog vartotojui kas kart nereiktų juos perrašinėti norint pakeisti įrašą.

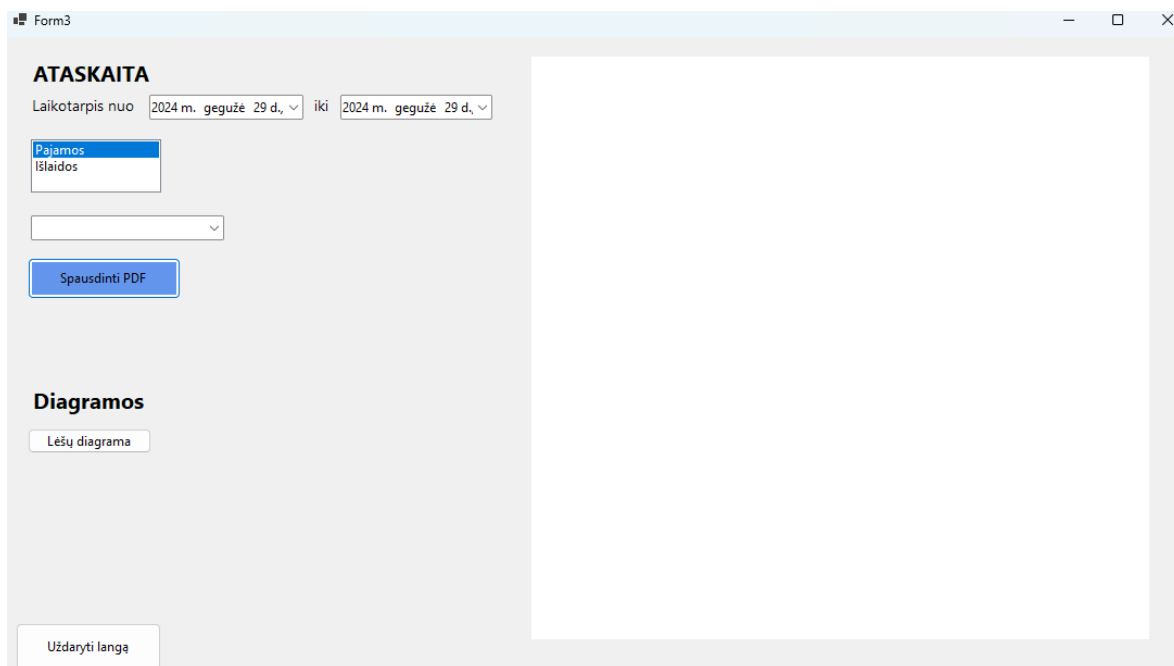
```

298 private void dataGridView1_CellContentClick(object sender, DataGridViewCellEventArgs e)
299 {
300     if (dataGridView1.SelectedRows.Count > 0)
301     {
302         DataGridViewRow selectedRow = dataGridView1.SelectedRows[0];
303
304         // Įrašyti į laukus
305         dateTimePicker1.Value = Convert.ToDateTime(selectedRow.Cells["Data"].Value);
306         sumos.Text = selectedRow.Cells["Suma"].Value.ToString();
307         textBox1.Text = selectedRow.Cells["Pastaba"].Value.ToString();
308         comboBox1.SelectedValue = selectedRow.Cells["Klasif_ID"].Value;
309         comboBox2.SelectedValue = selectedRow.Cells["Saltinis_ID"].Value;
310     }
311 }
312
313 private void button1_Click(object sender, EventArgs e)
314 {
315     var forma3 = new Form3();
316     forma3.Show();
317 }
318

```

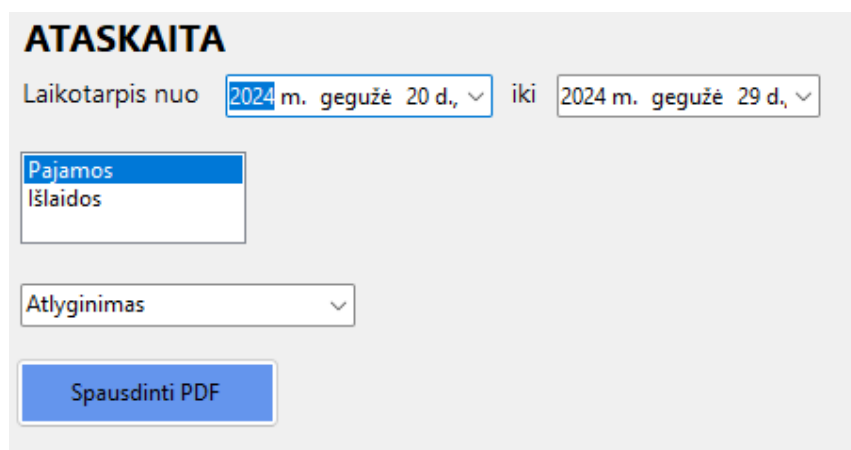
17 pav. Form2 kodas, duomenų įrašymas į laukus

Toliau eina trečia forma (aplinka), kurioje yra ataskaitų ir grafikų vaizdavimas. Šioje aplinkoje galima pasirinkti laikotarpį, kurį nori matyti vartotojas, lėšų tipą ar tai išlaidos ar pajamos ir kategoriją (alga, pramogos, kuras ir t.t.). Pasirinkus visus laukus spausti ant Spausdinti PDF ir paspaudus šį mygtuką vartotojui duodamas pasirinkimas, kur jis nori išsisaugoti savo pdf ataskaitą.



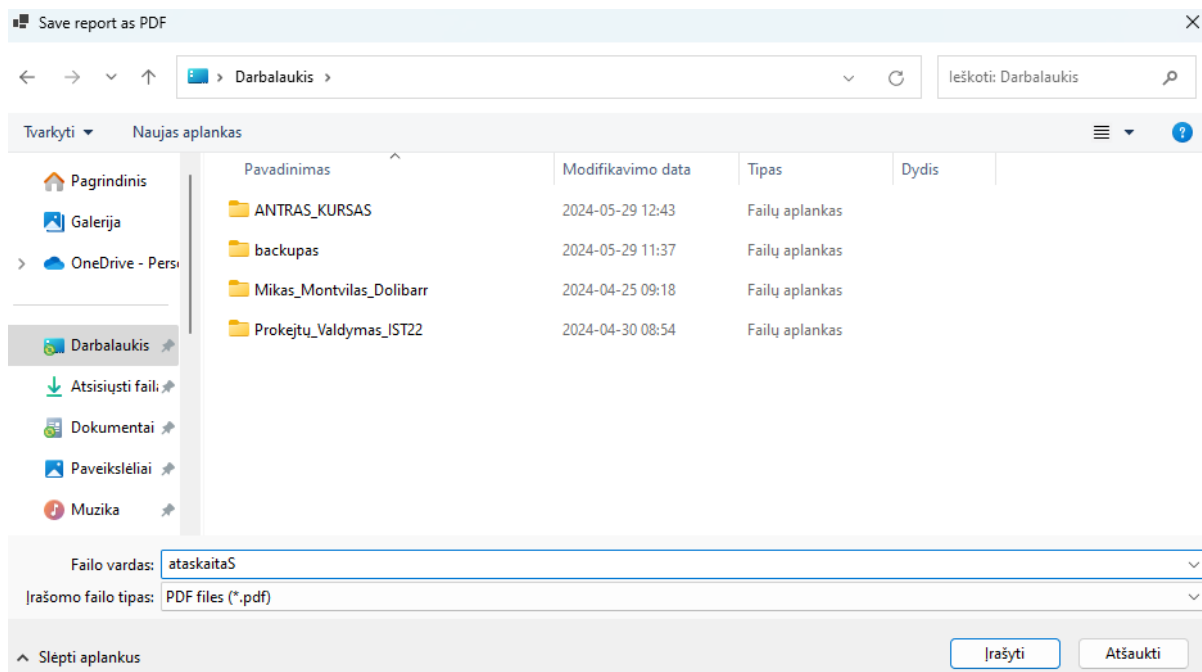
The screenshot shows a web application window titled 'Form3'. The main section is titled 'ATASKAITA'. It contains a date range selector with 'Laikotarpis nuo' followed by a date picker set to '2024 m. gegužė 29 d.' and 'iki' followed by another date picker set to '2024 m. gegužė 29 d.'. Below this is a dropdown menu with 'Pajamos' selected and 'Išlaidos' as an option. There is a 'Spausdinti PDF' button. Below the main form is a section titled 'Diagramos' with a 'Lėšų diagrama' button. At the bottom left is a 'Uždaryti langą' button. The right side of the window is a large empty white area.

18 pav. Form3 aplinka



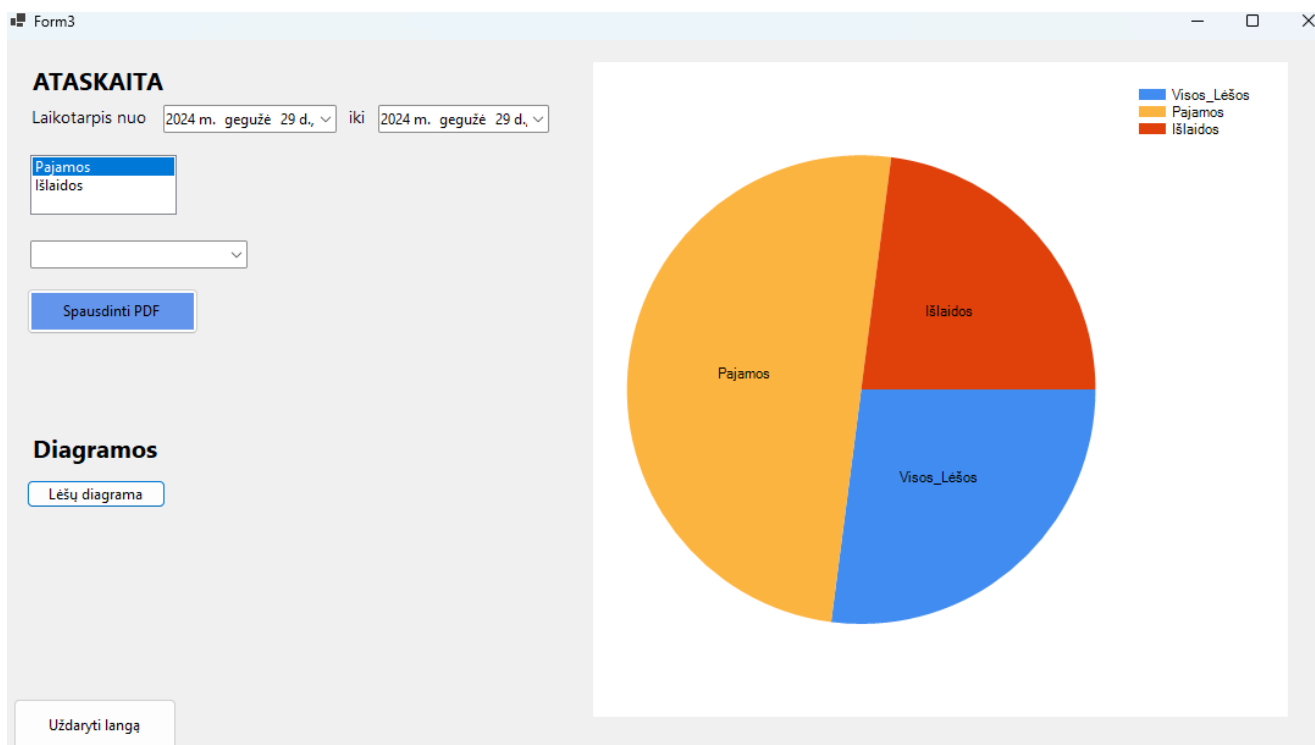
This is a close-up of the 'ATASKAITA' form. The date range is 'Laikotarpis nuo 2024 m. gegužė 20 d.' to 'iki 2024 m. gegužė 29 d.'. The dropdown menu shows 'Pajamos' selected and 'Išlaidos' as an option. Below it is a dropdown menu labeled 'Atlyginimas'. At the bottom is a 'Spausdinti PDF' button.

19 pav. Form3 aplinkos pasirinkimai



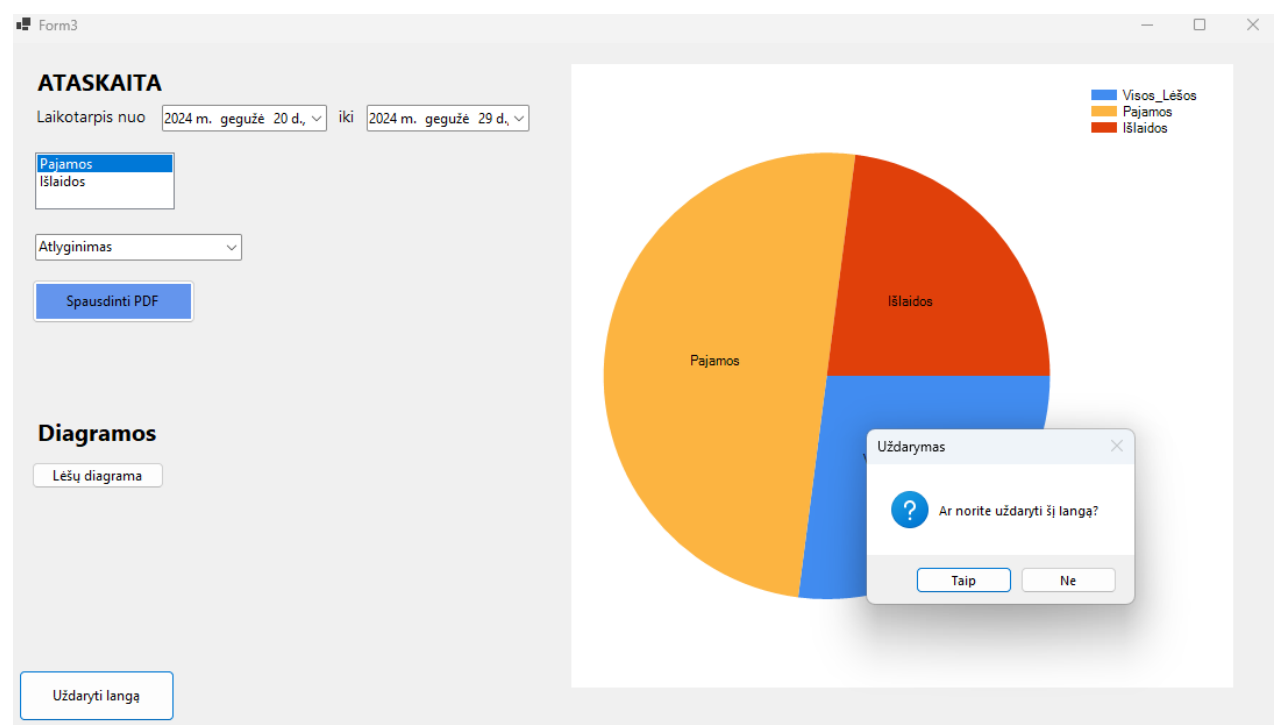
20 pav. Išsaugojimo laukas

Spaudžiant ant mygtuko Lėšų diagrama aplinkoje parodys lėšų diagramą, kurioje yra vaizduojama skritulinė diagrama, kuri parodo lėšų pasiskirstymą ar tai išlaidos, pajamos ar tiesiog visas sąskaitos likutis.



21 pav. Form3 aplinka, skritulinė diagrama

O taip atrodo šio lango uždarymas.



22 pav. Lango uždarymas

PDF ataskaita atrodo taip:

## Ataskaita

Laikotarpis nuo: 2024-05-20 iki: 2024-05-29

Kategorija: Atlyginimas

Tipas: Pajamos

Data	Suma	Pastaba	Saltinis
2024-05-24 08:31:49	1500,0000	Uždarb	Asmenin sskaita
2024-05-27 10:30:18	1500,0000	Darbas	Asmenin sskaita
2024-05-28 10:26:32	1500,0000	Už darb pas Oleg	Asmenin sskaita
2024-05-29 12:57:59	1500,0000	Gauta alga uz geguzes men.	Asmenin sskaita



Trečios formos kodas:

Šioje dalyje yra aprašytos visos naudojamos ir nenaudojamos trečioje formoje bibliotekos, čia taipogi yra aprašytas prisijungimas į duomenų bazę. Šioje dalyje irgi yra programos lauko uždarymas, kuris tiesiog uždaro trečios formos lauką.

```
1 using System;
2 using System.Data;
3 using System.Data.SqlClient;
4 using System.Drawing;
5 using System.IO;
6 using System.Windows.Forms;
7 using System.Windows.Forms.DataVisualization.Charting;
8 using iText.Kernel.Pdf;
9 using iText.Layout;
10 using iText.Layout.Element;
11 using iText.Layout.Properties;
12 using iText.IO.Image;
13
14 namespace Projektas
15 {
16     public partial class Form3 : Form
17     {
18         // Prisijungimas
19         public SqlConnection cnn;
20         string con = @"Data Source=localhost;Initial Catalog=Projektas;User ID=sa;Password=sa";
21         public Form3()
22         {
23             InitializeComponent();
24         }
25
26         1 reference
27         private void button1_Click(object sender, EventArgs e)
28         {
29             // Formos uždarymas
30             DialogResult ats;
31             ats = MessageBox.Show("Ar norite uždaryti šį langą?", "Uždarymas", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
32             if (ats == DialogResult.Yes)
33             {
34                 this.Close();
35             }
36     }
```

24 pav. Form3 kodas, lango uždarymas

Šioje dalyje yra aprašytas skritulinės diagramos kodas, visi duomenys yra pasiimami iš duomenų bazės su SELECT pagalbą ir jie toliau atvaizduojami skritulinėje diagramoje. Po SELECT užklausa apačioje yra aprašoma skritulinė diagrama, jos legenda, pavadinimas, dalių pavadinimai.

```
38 private void button2_Click(object sender, EventArgs e)
39 {
40     // Diagramos kūrimas, skritulinė diagrama
41     DataTable dt = new DataTable();
42
43     string query = @"SELECT SUM(Suma) AS Visos_Lėšos,
44                          SUM(CASE WHEN Suma > 0 THEN Suma ELSE 0 END) AS Pajamos,
45                          SUM(CASE WHEN Suma < 0 THEN Suma ELSE 0 END) AS Išlaidos FROM Pinigai";
46
47     SqlDataAdapter da = new SqlDataAdapter(query, con);
48
49     da.Fill(dt);
50
51     chart1.DataSource = dt;
52
53     chart1.Series.Clear();
54     Series series = chart1.Series.Add("Pinigai");
55     series.ChartType = SeriesChartType.Pie;
56
57     foreach (DataRow row in dt.Rows)
58     {
59         series.Points.AddXY("Visos_Lėšos", row["Visos_Lėšos"]);
60         series.Points.AddXY("Pajamos", row["Pajamos"]);
61         series.Points.AddXY("Išlaidos", row["Išlaidos"]);
62     }
63
64     chart1.DataBind();
65 }
```

25 pav. Form3 kodas, skritulinė diagrama

Šioje dalyje yra parašytas kodas, toks kai yra užkraunama trečia formą į pasirinkimus iš duomenų bazės lentelės su SELECT pagalbą yra įkeliami pasirinkimai į formos elementus t.y. lėšos, pajamos, alga, pramogos ir t.t.

```
82 1 reference private void Form3_Load(object sender, EventArgs e)
83  {
84      SqlConnection connection = new SqlConnection(con);
85      {
86          connection.Open();
87
88          // Kategorija ComboBox (comboBox1)
89          string queryKategorija = "SELECT ID, Pavadinimas FROM Klasifikatorius";
90          SqlCommand cmd = new SqlCommand(queryKategorija, connection);
91          SqlDataReader reader = cmd.ExecuteReader();
92
93          DataTable kategorijaTable = new DataTable();
94          kategorijaTable.Load(reader);
95          comboBox1.DataSource = kategorijaTable;
96          comboBox1.DisplayMember = "Pavadinimas";
97          comboBox1.ValueMember = "ID";
98
99
100         // Tipas ListBox1 (listBox1)
101         string queryTipas = "SELECT ID, Tipo_pav, Daugiklis FROM Tipas";
102         SqlCommand cmdas = new SqlCommand(queryTipas, connection);
103         SqlDataReader readeras = cmdas.ExecuteReader();
104
105         DataTable TipasTable = new DataTable();
106         TipasTable.Load(readeras);
107         listBox1.DataSource = TipasTable;
108         listBox1.DisplayMember = "Tipo_pav";
109         listBox1.ValueMember = "ID";
110
111         listBox1.SelectedIndexChanged += listBox1_SelectedIndexChanged;
112         comboBox1.DataSource = null;
113     }
114 }
```

26 pav. Form3 kodas, formos užkrovimas

Šioje dalyje yra toks pat kodas kaip ir antroje formoje su lėšų pasirinkimais, jog vartotojas matytų logiškus pasirinkimus pagal prieš tai pasirinktą lėšų tipą.

```
116 private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
117 {
118     if (listBox1.SelectedValue != null)
119     {
120         DataRowView selectedRow = listBox1.SelectedItem as DataRowView;
121         if (selectedRow != null)
122         {
123             int selectedTipasId = Convert.ToInt32(selectedRow["ID"]);
124
125             SqlConnection connection = new SqlConnection(con);
126             {
127                 connection.Open();
128
129                 // Query gauti geras kategorijas pagal pasirinktą tipą (pajamos ar išlaidos)
130                 string queryKategorija = "SELECT ID, Pavadinimas FROM Klasifikatorius WHERE Tipas_ID = @TipasID";
131                 SqlCommand cmd = new SqlCommand(queryKategorija, connection);
132                 {
133                     cmd.Parameters.AddWithValue("@TipasID", selectedTipasId);
134                     SqlDataReader reader = cmd.ExecuteReader();
135                     {
136                         DataTable kategorijaTable = new DataTable();
137                         kategorijaTable.Load(reader);
138                         comboBox1.DataSource = kategorijaTable;
139                         comboBox1.DisplayMember = "Pavadinimas";
140                         comboBox1.ValueMember = "ID";
141                     }
142                 }
143             }
144         }
145     }
146 }
```

27 pav. Form3 kodas, pasirinkimų keitimas

Šioje dalyje parašytas kodas PDF saugojimui, jog vartotojui pateikus prieš tai nurodytus kriterijus jis pats galėtų išsirinkti savo pdf ataskaitos išsaugojimo vietą.

```
158 private void GavimasPDF_Click(object sender, EventArgs e)
159 {
160     // PDF saugojimas rankiniu būdu
161     SaveFileDialog saveFileDialog = new SaveFileDialog();
162     saveFileDialog.Filter = "PDF files (*.pdf)|*.pdf";
163     saveFileDialog.Title = "Save report as PDF";
164     if (saveFileDialog.ShowDialog() == DialogResult.OK)
165     {
166         GeneratePdfReport(saveFileDialog.FileName);
167     }
168 }
```

28 pav. Form3 kodas, pdf saugojimas

Šita kodo dalis aprašo pdf spausdinimą taip kaip atrodys informacija pdf faile. Čia tiesiog sukuriamas pdf failas aprašomi šriftai ir teksto buvimo vietos, toliau eina SQL užklausa, kuri įkelia duomenis iš duomenų bazės lentelės, bet aišku pagal prieš tai vartotojo pasirinktais kriterijais. Kodo apačioje taipogi yra ir foreach ciklas, kuris atspausdina lentelę, kuri matoma pdf faile.

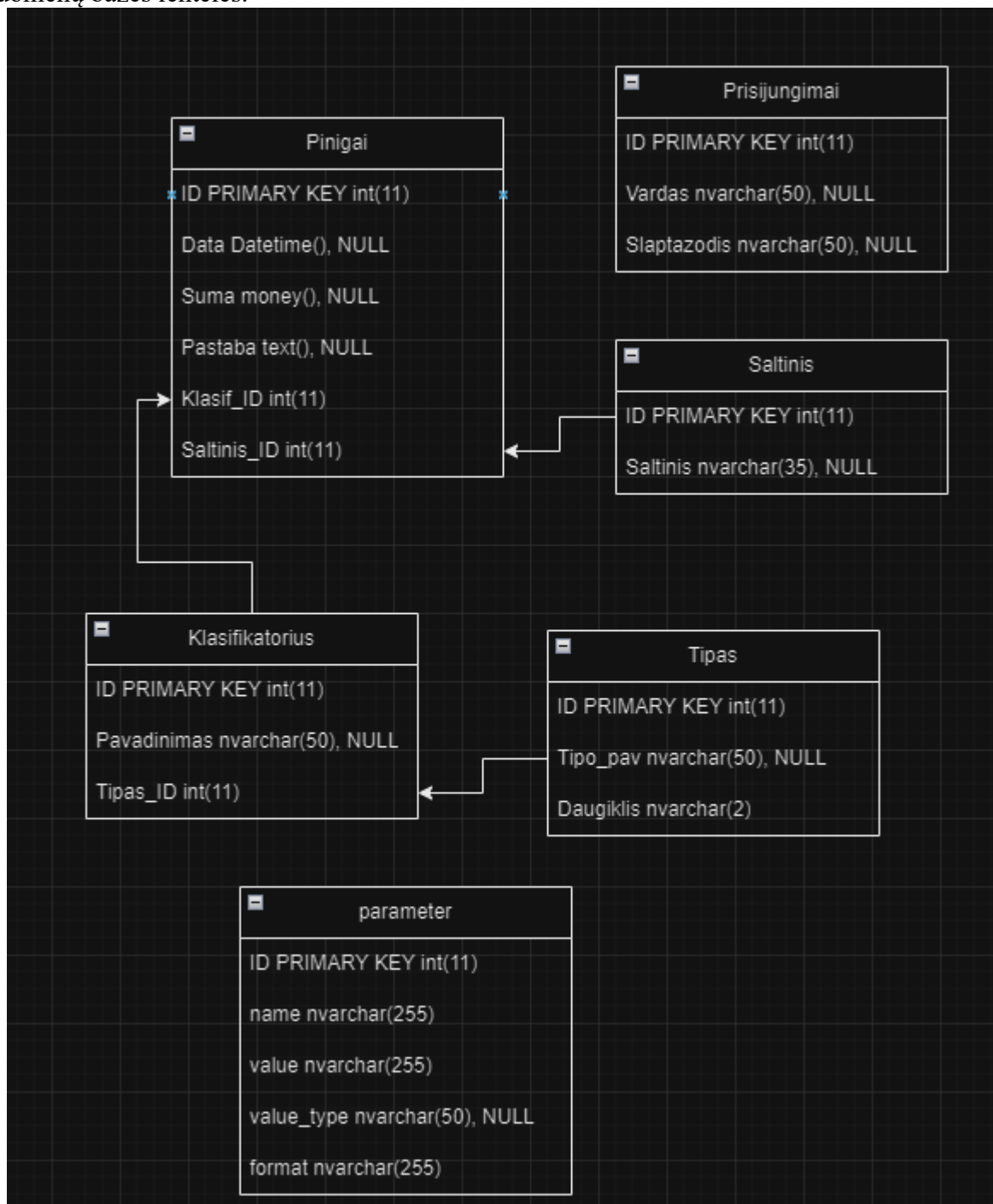
```

170 private void GeneratePdfReport(string filePath)
171 {
172     PdfWriter writer = new PdfWriter(filePath);
173     PdfDocument pdf = new PdfDocument(writer);
174     Document document = new Document(pdf);
175
176     // Failo header
177     Paragraph title = new Paragraph("Ataskaita")
178         .SetTextAlignment(iText.Layout.Properties.TextAlignment.CENTER)
179         .SetFontSize(20);
180     document.Add(title);
181
182     // Ataskaitos laikotarpis
183     Paragraph dateRange = new Paragraph($"Laikotarpis nuo: {dateTimePicker2.Value.ToShortDateString()} iki: {dateTimePicker3.Value.ToShortDateString()}")
184         .SetTextAlignment(iText.Layout.Properties.TextAlignment.LEFT)
185         .SetFontSize(12);
186     document.Add(dateRange);
187
188     // Kategorija
189     Paragraph category = new Paragraph($"Kategorija: {comboBox1.Text}")
190         .SetTextAlignment(iText.Layout.Properties.TextAlignment.LEFT)
191         .SetFontSize(12);
192     document.Add(category);
193
194     // Tipas
195     Paragraph type = new Paragraph($"Tipas: {listBox1.Text}")
196         .SetTextAlignment(iText.Layout.Properties.TextAlignment.LEFT)
197         .SetFontSize(12);
198     document.Add(type);
199
200     // SQL užklausa
201     string query = @" SELECT Data, Suma, Pastaba, Saltinis.Saltinis FROM Pinigai JOIN Saltinis ON Pinigai.Saltinis_ID = Saltinis.ID WHERE Data BETWEEN @s";
202
203     DataTable dt = new DataTable();
204     SqlConnection connection = new SqlConnection(con);
205     SqlCommand cmd = new SqlCommand(query, connection);
206     cmd.Parameters.AddWithValue("@startDate", dateTimePicker2.Value);
207     cmd.Parameters.AddWithValue("@endDate", dateTimePicker3.Value);
208     cmd.Parameters.AddWithValue("@categoryID", comboBox1.SelectedValue);
209
210     SqlDataAdapter da = new SqlDataAdapter(cmd);
211     da.Fill(dt);
212
213
214     // Pridedama į PDF
215     Table table = new Table(UnitValue.CreatePercentArray(new float[] { 1, 2, 2, 2 })).UseAllAvailableWidth();
216     table.AddHeaderCell("Data");
217     table.AddHeaderCell("Suma");
218     table.AddHeaderCell("Pastaba");
219     table.AddHeaderCell("Saltinis");
220
221     foreach (DataRow row in dt.Rows)
222     {
223         table.AddCell(new Cell().Add(new Paragraph(row["Data"].ToString())));
224         table.AddCell(new Cell().Add(new Paragraph(row["Suma"].ToString())));
225         table.AddCell(new Cell().Add(new Paragraph(row["Pastaba"].ToString())));
226         table.AddCell(new Cell().Add(new Paragraph(row["Saltinis"].ToString())));
227     }
228
229     document.Add(table);
230
231     document.Close();
232
233 }

```

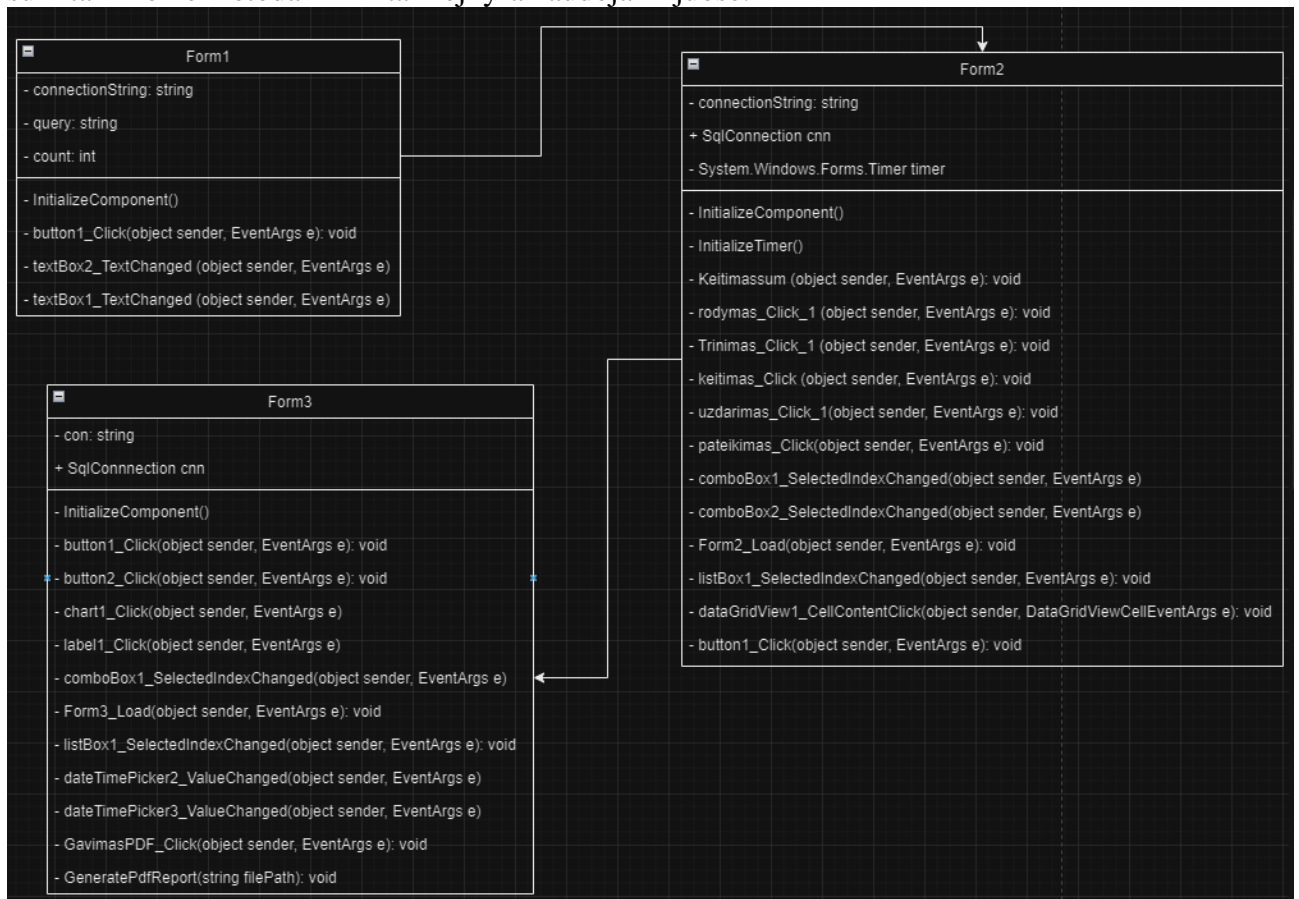
29 pav. Form3 kodas, PDF kūrimas

Štai kaip atrodo duomenų bazės UML diagrama ir kaip lentelės yra susijusios. Lentelių yra šešios prisijungimų ir parameter lentelės nėra susijusios su pagrindinėmis lentelėmis, jos tiesiog atlieka pagalbines funkcijas kaip leidžia vartotojui prisijungti ir įkelti/saugoti parametrus iš duomenų bazės lentelės.



30 pav. Duomenų bazės diagrama

O čia yra UML diagrama padaryta pagal sukurtą kodą ir kaip formos sąveikauja viena su kita ir kokie metodai ir kintamieji yra naudojami juose.



31 pav. Projekto UML diagrama