

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**(РУТ (МИИТ)**

**ИНСТИТУТ ТРАНСПОРТНОЙ ТЕХНИКИ И СИСТЕМ УПРАВЛЕНИЯ**

**Лабораторная работа №5**

**по дисциплине** «**Методы программирования**»

«Автоматизация Word»

**Выполнили:** ст. гр. ТКИ-341

Поваляева А.В.

Топталова С.Д.

**Проверил:** к.т.н. Сафронов А.И.

**Москва – 2024 г**

1. **Цель работы**

Освоить навыки подключения внешних модулей (библиотек классов) и их использования в разрабатываемых программах; освоить навыки программируемой настройки параметров документов *Microsoft Office Word*.

## 2. Формулировка задачи

1.  Создать и настроить графический пользовательский интерфейс, взаимодействующий с текстовым файлом (*\*.txt / \*.csv*), а также с модулем (библиотекой классов) текстового редактора *Microsoft Office Word*.

2.  Создать текстовый файл. Текстовый файл исходно пуст, но нацелен на хранение *M* строк с разделителями вида:

|  |  |
| --- | --- |
| *param1;param2;param3;…;paramN* | (1) |

Каждая строка – набор значимых параметров, записанных в порядке, предусмотренных автором-разработчиком графического пользовательского интерфейса. Количество параметров *N* так же определяется автором-разработчиком.

Количество строк *M* определяется количеством абзацев на титульном листе отчёта по выполненной работе в формате *Microsoft Office Word*. Например:

а. Наименование ведомства (Министерство транспорта Российской Федерации);

б. Регалии вуза (Федеральное государственное автономное образовательное учреждение высшего образования);

в. Наименование вуза («Российский университет транспорта» (РУТ (МИИТ));

г. Наименование института (Институт транспортной техники и систем управления);

д. Наименование кафедры (Кафедра «Управление и защита информации»);

е. Вид документа (Отчёт / Реферат / Курсовая работа / Курсовой проект);

ж. Вариация связки (по практике / по);

и. Вид учебного занятия (Учебной практике / Производственной практике / Лабораторной работе / Практическому занятию / Индивидуальному заданию);

к. Вариация связки, если необходима (по дисциплине);

л. Наименование дисциплины, если необходимо («Алгоритмизация и технологии программирования» / «Методы программирования»);

м. Вариация связки, если необходима (на тему / представлено решение задачи);

н. Строка «Выполнил: ст. гр.» с указанием номера учебной группы;

о. Ф.И.О. автора-составителя документа;

п. Номер варианта;

р. Строка «Проверил: » с указанием регалий и Ф.И.О. проверяющего лица.

с. Строка формата «Город – год».

Например, *param1* – строка текста, выводимая в абзац документа *Microsoft Office Word; param2* – семейство/наименование шрифта (*Times New Roman*); *param3* – размер шрифта (15 пт) и так далее.

3.  При загрузке графического пользовательского интерфейса все параметры из файла считываются в строковый массив, если файл не пуст, и изменяют состояние интерфейсных элементов управления, размещённых на графическом пользовательском интерфейсе. Если файл пуст, то все интерфейсные элементы управления графического пользовательского интерфейса остаются в исходном, нулевом состоянии.

4.  Должна быть предусмотрена экранная кнопка «Сохранить», записывающая в файл выставленные или изменённые на графическом пользовательском интерфейсе значения параметров.

5.  Должна быть предусмотрена кнопка «Создать», инициирующая генерацию титульного листа в формате *Microsoft Office Word.*

6.  Должен быть предусмотрен предварительный просмотр документа, планируемого к выгрузке (рассмотреть вариант использования интерфейсного элемента *PictureBox* для этих целей).

7.  Документ, представленный иллюстрацией, должен быть первично набран в *Microsoft Office Word* вручную. Он запланирован в качестве тестового примера в данной работе.

8.  Все заполнители в работе выполняются при помощи «кареток»

[, , ] (*Tab Stops*) и табуляции [https://prosto-kod.ru/17_01_2023/4.jpg] (*Tab*).

9.  Все неявные колоночные выравнивания допустимо выполнять необрамлёнными таблицами (таблицами без визуальных границ).

## Диаграммы классов, входящих в состав решения.

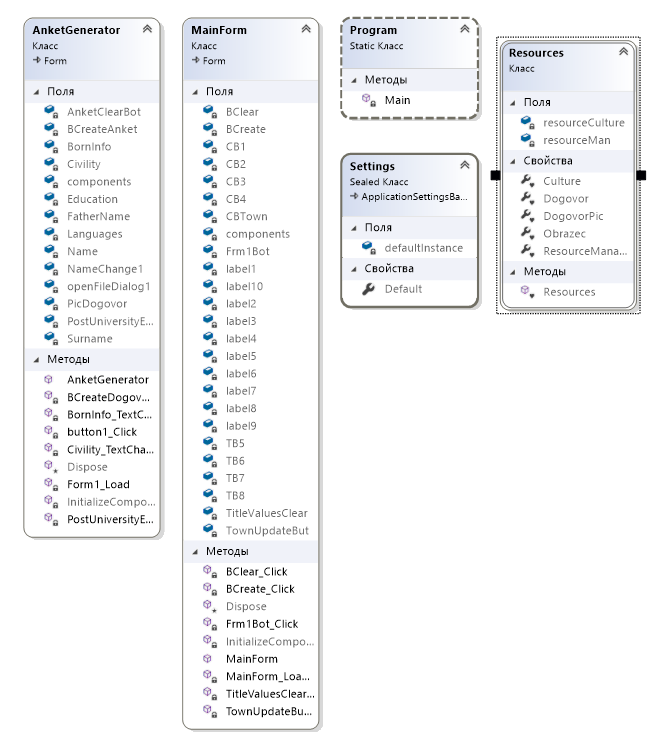


Рисунок 1 – Диаграмма классов

## Сеть Петри

**Легенда:**



Рисунок 2 – Значения обозначений

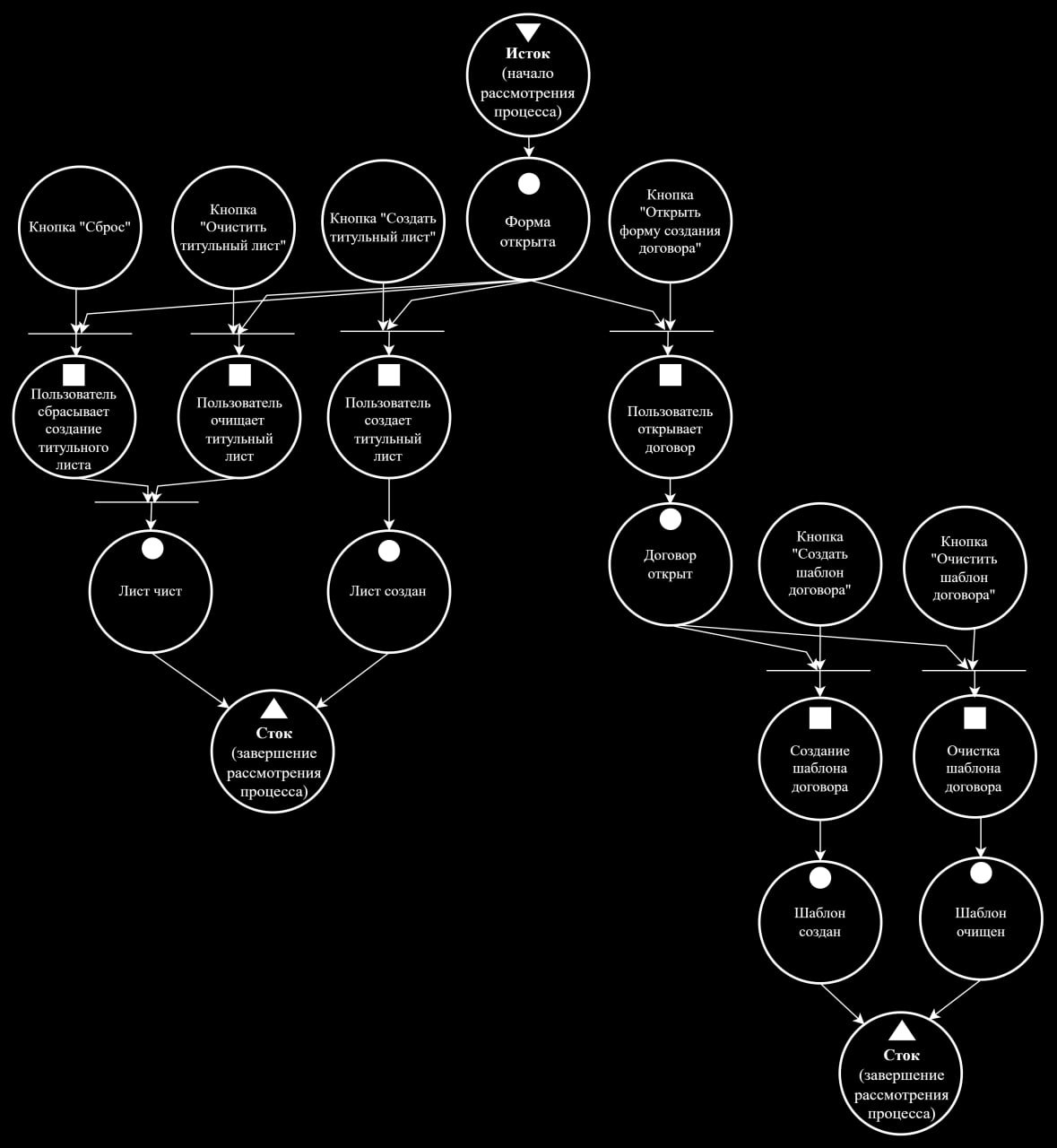


Рисунок 3 – Сеть Петри

## Составление схем алгоритмов методов в составе решения, отмеченных на сети Петри в качестве «эффектов» (метка )

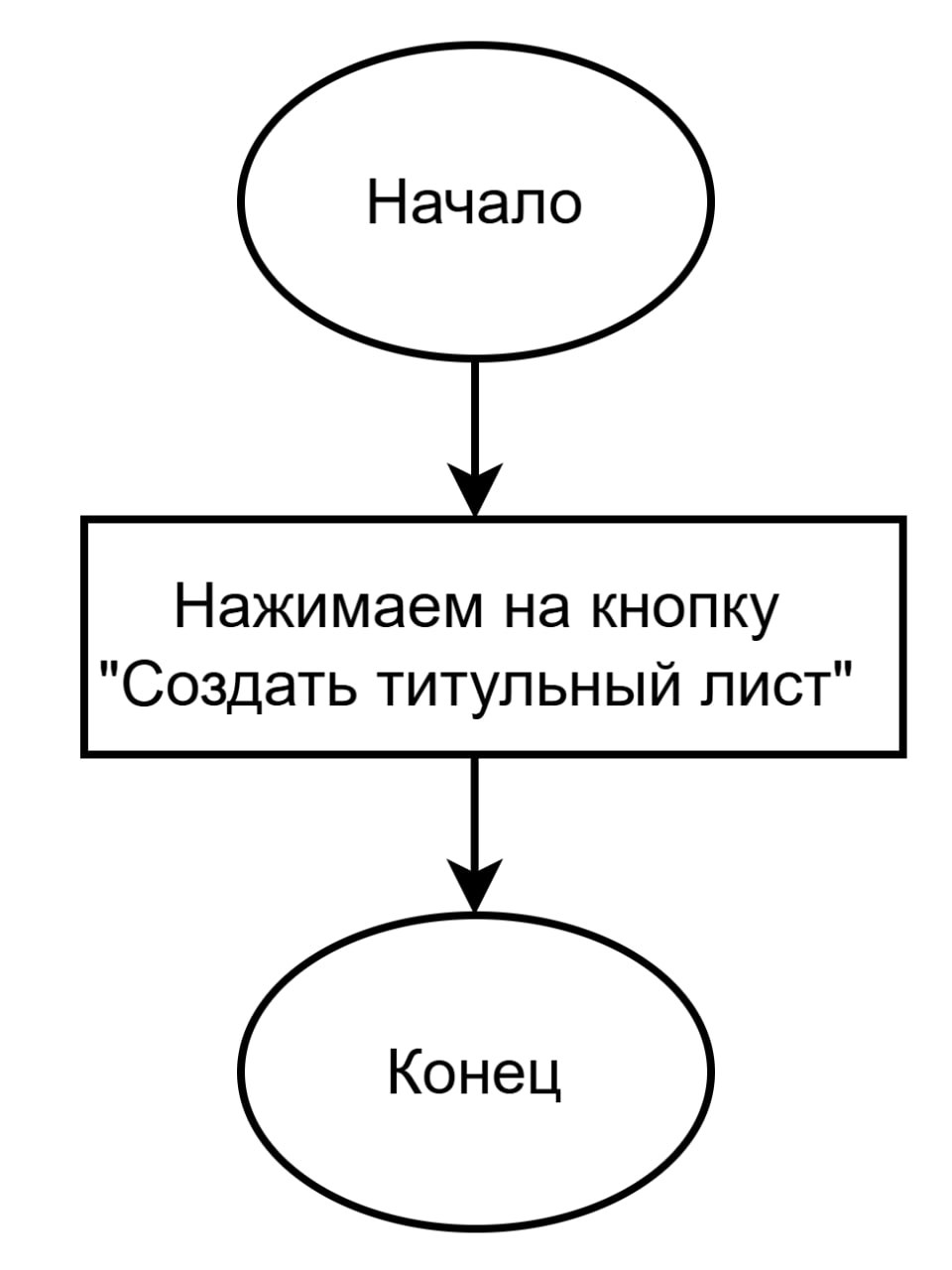


Рисунок 4 – Создаем шаблон титульного листа

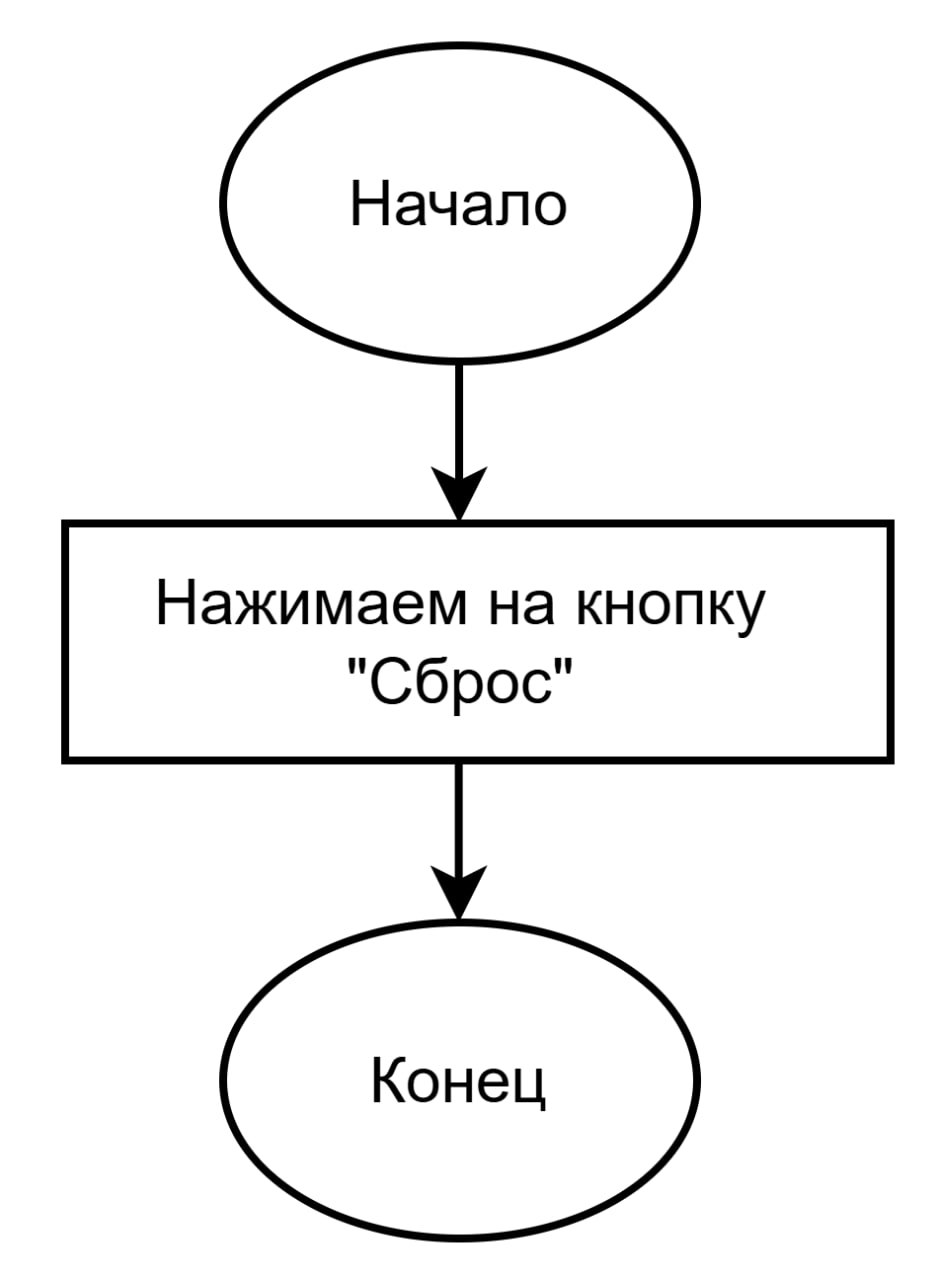


Рисунок 5 – Очищаем титульный лист

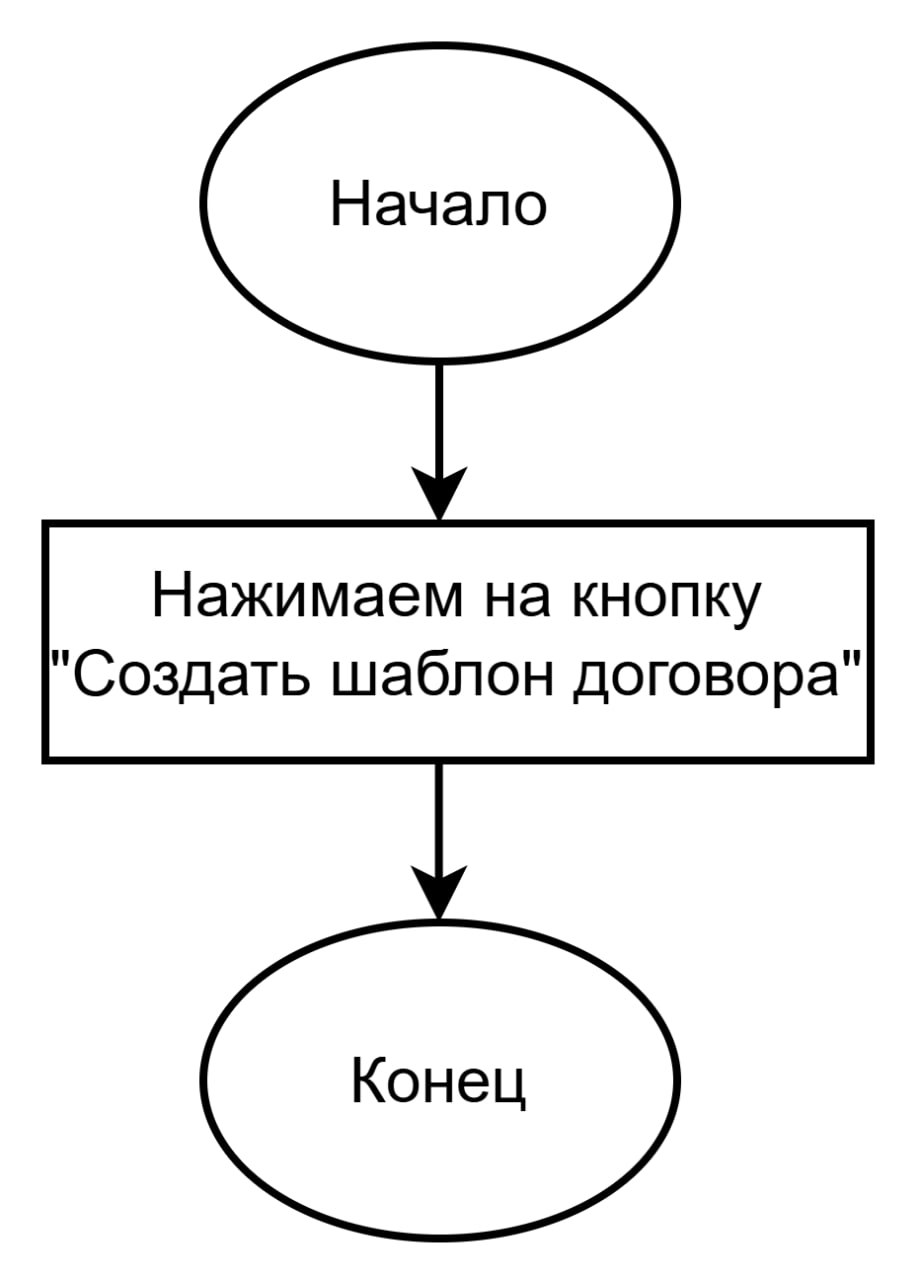


Рисунок 6 – Создаем шаблон договора

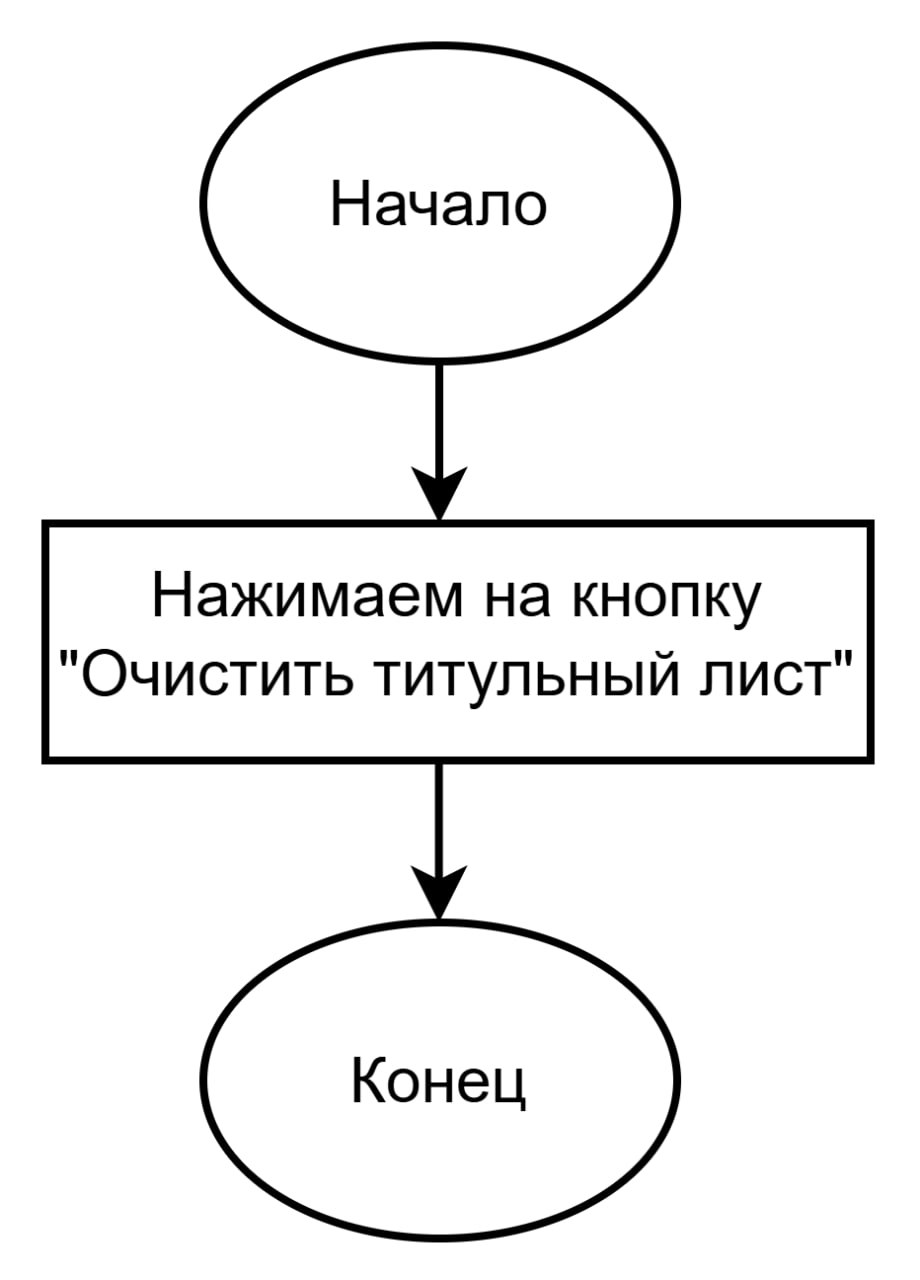


Рисунок 7 – Очистка настроек шаблона титульного листа

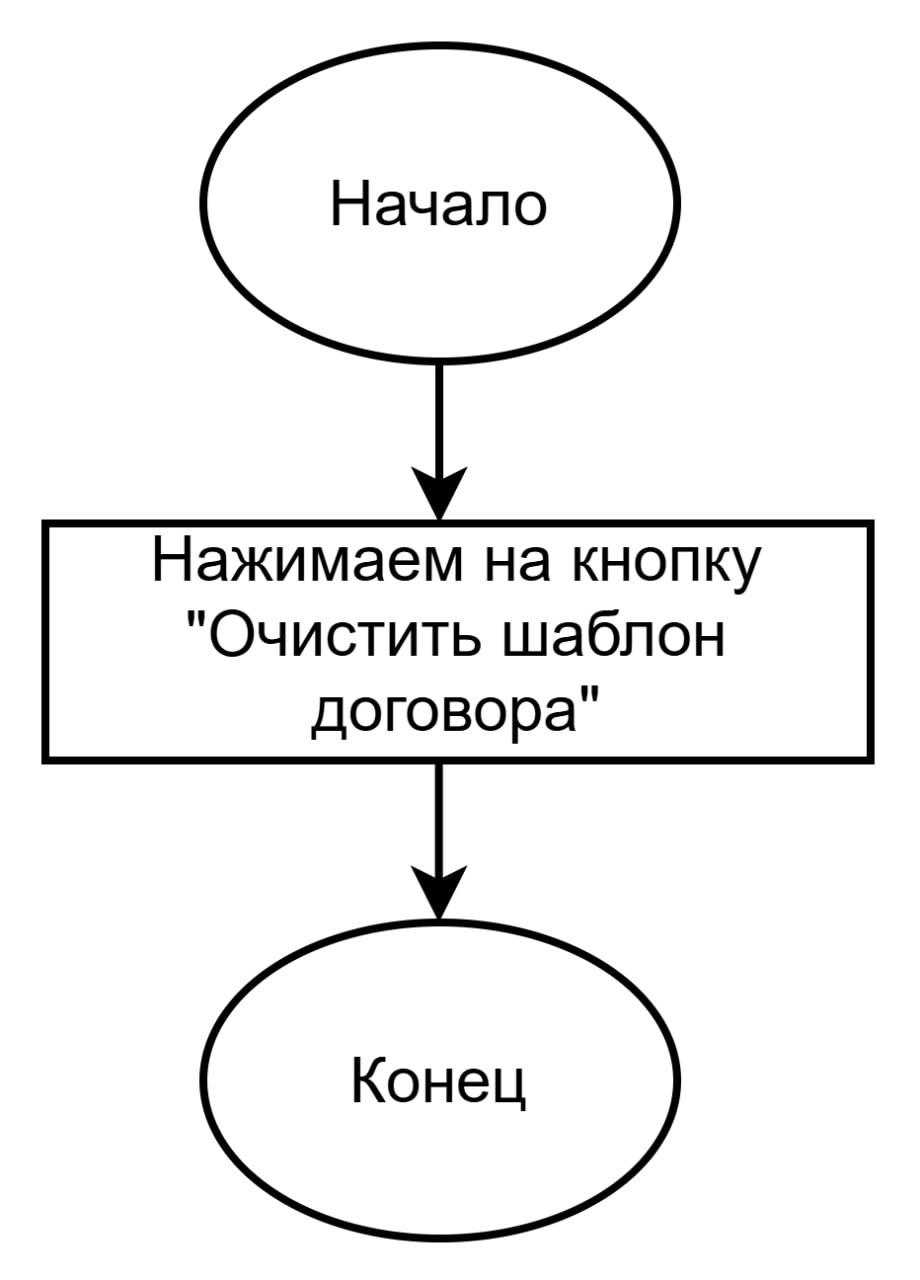


Рисунок 8 – Очищаем договор

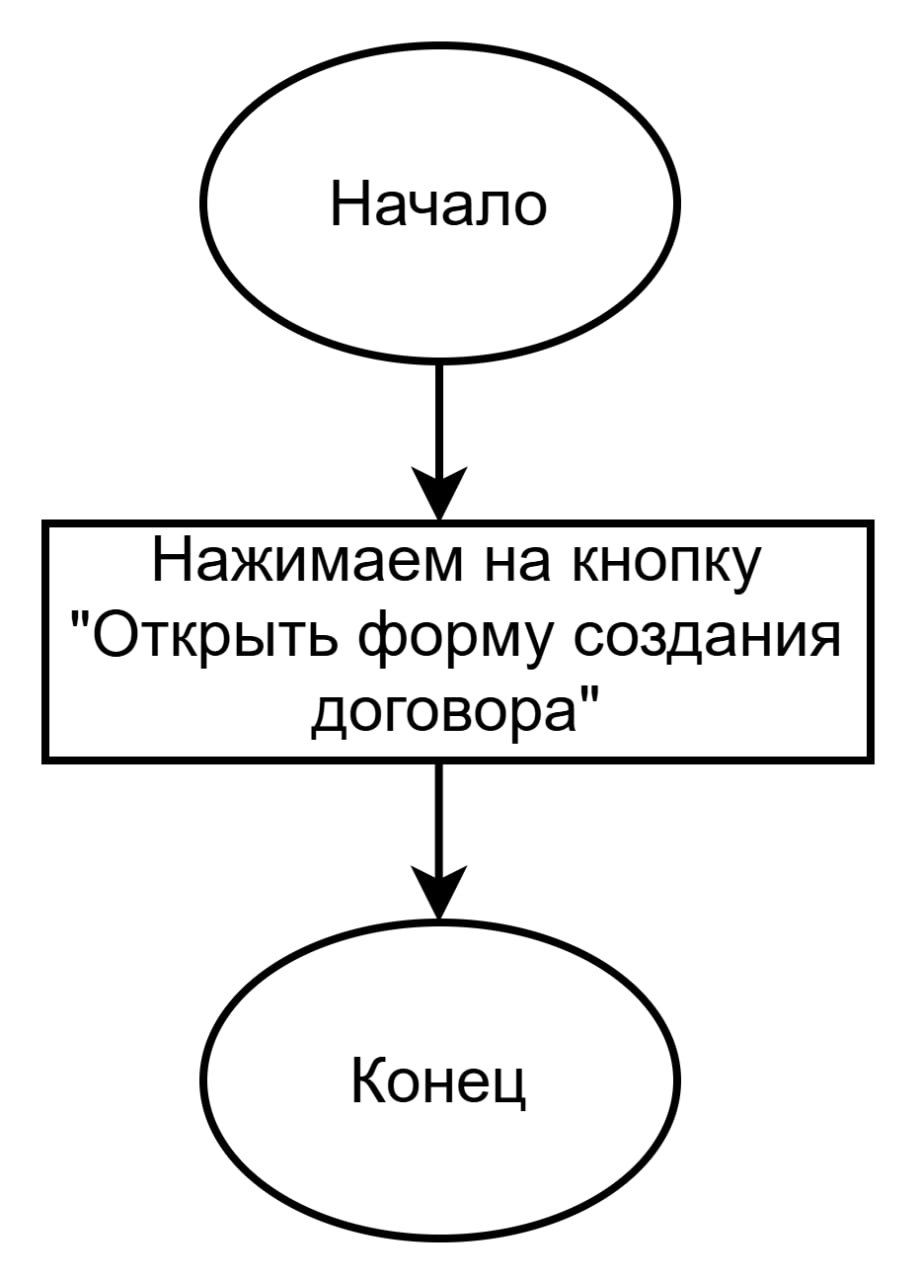


Рисунок 9 – Открываем договор

## Листинг (код) составленного программного обеспечения Основная форма:

using System;

using System.Collections.Generic;

using System.IO;

using System.Text;

using System.Windows.Forms;

using Word = Microsoft.Office.Interop.Word;

namespace Wordik

{

public partial class MainForm : Form

{

public MainForm()

{

InitializeComponent();

IList<String> Name = new List<String>();

CB1.Items.AddRange(new string[] { "к.т.н. Сафронов А.И.", "к.т.н. Кулагин М.А." });

CB2.Items.AddRange(new string[] { "Отчёт", "Реферат", "Эссе", "Курсовой проект", "Курсовая работа", "Доклад", "Домашнее задание" });

CB3.Items.AddRange(new string[] { "лабораторной работе", "практической работе", "индивидуальному заданию", "учебной практике", "производственной практике", "преддипломной практике" });

CB4.Items.AddRange(new string[] { "№1", "№2", "№3", "№4", "№5", "№6", "№7", "№8", "№9", "№10" });

this.FormBorderStyle = FormBorderStyle.FixedSingle;

}

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

{

Word.Application oWord;

Word.Document oDoc;

Word.Paragraph oPar;

string path = Application.StartupPath + "\\..\\.." + "\\NewDocument.docx";

DateTime now = DateTime.Today;

int year = now.Year;

string line = new string('\_', 20);

oWord = new Word.Application();

oDoc = oWord.Documents.Add();

Word.Range Range0 = oDoc.Content.Paragraphs.Last.Range;

Range0.Text = $"ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ\n" +

$"УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ\n" +

$"«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»\n" +

$"(РУТ (МИИТ)\n" +

$"\n" +

$"ИНСТИТУТ ТРАНСПОРТНОЙ ТЕХНИКИ И СИСТЕМ УПРАВЛЕНИЯ\n" +

$"\n {line}"+ $"\n" +

$"Кафедра «Управление и защита информации»";

Range0.Font.Size = 14;

Range0.Bold = 1;

Range0.Font.Name = "Times New Roman";

Range0.ParagraphFormat.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

// Получаем выбранные значения из комбобоксов

string Prep = CB1.SelectedItem.ToString();

string documentType = CB2.SelectedItem.ToString();

string workType = CB3.SelectedItem.ToString();

string grade = CB4.SelectedItem.ToString();

// Получаем значение из текстбокса

string topic = TB5.Text;

string nameDiscip = TB6.Text;

string GrStud = TB7.Text;

string FIOStud = TB8.Text;

Word.Range range3 = oDoc.Content.Paragraphs.Last.Range;

range3.Collapse(Word.WdCollapseDirection.wdCollapseEnd);

range3.Text = $"\n\n\n {documentType} по {workType} {grade}\n";

Word.Range Range1 = oDoc.Content.Paragraphs.Last.Range;

Range1.Start = range3.Start;

Range1.End = range3.End;

Range1.Font.Size = 16; // Размер шрифта

Range1.ParagraphFormat.SpaceAfter = 1; // Междустрочный интервал

Range1.Font.Bold = 1; // Сделать текст жирным (1 - жирный, 0 - обычный)

Word.Range range4 = oDoc.Content.Paragraphs.Last.Range;

range4.Collapse(Word.WdCollapseDirection.wdCollapseEnd);

range4.Text = $"\n«{topic}»\n\n\n\n\n\n\n";

// Форматирование новыми строками

Word.Range addedRange1 = oDoc.Content.Paragraphs.Last.Range;

addedRange1.Start = range4.Start;

addedRange1.End = range4.End;

addedRange1.Font.Size = 14; // Размер шрифта

addedRange1.Font.Bold = 0; // Сделать текст жирным (1 - жирный, 0 - обычный)

// Получаем последний параграф в документе

Word.Range lastRange = oDoc.Content.Paragraphs.Last.Range;

lastRange.ParagraphFormat.Alignment = Word.WdParagraphAlignment.wdAlignParagraphRight;

lastRange.Text = $"Выполнил: ст. гр. {GrStud}\n" +

$"{FIOStud}\n" +

$"Проверил: {Prep}\n\n\n";

lastRange.Font.Size = 14;

lastRange.Font.Name = "Times New Roman";

Word.Range footerParagraph = oDoc.Content.Paragraphs.Last.Range;

footerParagraph.Text = $"{CBTown.SelectedItem} – {year}";

footerParagraph.ParagraphFormat.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

footerParagraph.Font.Bold = 1;

footerParagraph.Font.Color = Word.WdColor.wdColorBlack;

footerParagraph.Font.ColorIndex = Word.WdColorIndex.wdBlack;

//Сохраняем документ

try

{

oDoc.SaveAs(path);

}

catch (Exception ex)

{

MessageBox.Show("Ошибка при чтении файла: " + ex.Message, "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

oWord.Visible = true;

}

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

{

Word.Application oWord;

Word.Document oDoc;

string path = Application.StartupPath + "\\NewDocument.docx";

oWord = new Word.Application();

oDoc = oWord.Documents.Add();

// Создаем таблицу с 4 строками и 2 столбцами

int numRows = 4;

int numCols = 2;

Word.Table table = oDoc.Tables.Add(oDoc.Range(0, 0), numRows, numCols);

// Настройка ширины столбцов для заполнения всего листа

table.AutoFitBehavior(Word.WdAutoFitBehavior.wdAutoFitWindow);

// Настройка интервалов для каждой строки

table.Rows[1].Height = 200f; // Устанавливаем интервал для первой строки

table.Rows[2].Height = 200f; // Устанавливаем интервал для второй строки

table.Rows[3].Height = 260f; // Устанавливаем интервал для третьей строки

table.Rows[4].Height = 60f; // Устанавливаем интервал для четвертой строки

/// Объединяем ячейки в указанных строках

table.Cell(1, 1).Merge(table.Cell(1, 2)); // Объединяем ячейки в первой строке

table.Cell(2, 1).Merge(table.Cell(2, 2)); // Объединяем ячейки во второй строке

table.Cell(4, 1).Merge(table.Cell(4, 2)); // Объединяем ячейки в четвертой строке

// Делаем всю таблицу невидимой

foreach (Word.Cell c in table.Range.Cells)

{

c.Shading.BackgroundPatternColorIndex = Word.WdColorIndex.wdNoHighlight;

c.Borders.Enable = 0;

}

oDoc.SaveAs(path);

oWord.Visible = true;

}

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

{

CB1.SelectedIndex = -1;

CB2.SelectedIndex = -1;

CB3.SelectedIndex = -1;

CB4.SelectedIndex = -1;

CBTown.SelectedIndex = -1;

TB5.Clear();

TB6.Clear();

TB7.Clear();

TB8.Clear();

}

private void Frm1Bot\_Click(object sender, EventArgs e)

{

AnketGenerator TitleEdit = new AnketGenerator();

TitleEdit.ShowDialog();

}

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

{

CBTown.Items.Clear();

try

{

StreamReader Reader = new StreamReader(Directory.GetCurrentDirectory() + "\\Towns.txt", Encoding.GetEncoding(1251));

string content = Reader.ReadToEnd();

string[] towns = content.Split(';');

foreach (string s in towns)

{

//TB5.Text = s;

CBTown.Items.Add(s);

}

Reader.Close();

}

catch (Exception ex)

{

MessageBox.Show("Ошибка при чтении файла: " + ex.Message, "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

}

## Форма анкеты:

using System;

using System.Windows.Forms;

using Word = Microsoft.Office.Interop.Word;

namespace Wordik

{

public partial class AnketGenerator : Form

{

public AnketGenerator()

{

InitializeComponent();

this.FormBorderStyle = FormBorderStyle.FixedSingle;

}

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

{

Word.Application oWord;

Word.Document oDoc;

Word.Paragraph oPar;

Word.Table oTable;

string path = System.Windows.Forms.Application.StartupPath + "\\..\\.." + "\\Anketa.docx";

oWord = new Word.Application();

oDoc = oWord.Documents.Add();

oPar = oDoc.Paragraphs.Add();

// Добавляем текст анкеты последовательного отображения

oPar.Range.Font.Name = "Times New Roman";

oPar.Range.Font.Size = 12;

oPar.Range.Font.Bold = 0;

oPar.Range.Text = "Утверждена \vРаспоряжением Правительства" +

$"\vРоссийской Федерации" +

$"\vот 26 мая 2005г. №667-р" +

$"\vв ред. от 5 марта 2018г.";

oPar.Alignment = Word.WdParagraphAlignment.wdAlignParagraphRight;

oPar.Range.InsertParagraphAfter();

oPar.Range.Font.Size = 14;

oPar.Range.Font.Bold = 1;

oPar.Range.Font.Name = "Times New Roman";

oPar.Range.Text = "АНКЕТА";

oPar.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

oPar.Range.InsertParagraphAfter();

oPar.Range.Font.Name = "Times New Roman";

oPar.Range.Font.Size = 12;

oPar.Range.Font.Bold = 0;

oPar.Alignment = Word.WdParagraphAlignment.wdAlignParagraphLeft;

oPar = oDoc.Paragraphs.Add();

oPar.Range.Text = $"1. Фамилия: {Surname.Text}" +

$"\n Имя: {UserName.Text}" +

$"\n Отчество: {FatherName.Text}";

oPar.Range.InsertParagraphAfter();

// Добавляем таблицу с вопросами

oTable = oDoc.Tables.Add(oPar.Range, 6, 2);

//Проставка в ячейки таблицы значений

oTable.Cell(1, 1).Range.Text = "2. Если изменяли фамилию, имя или отчество, то укажите их, а также когда, где и по какой причине изменяли";

oTable.Cell(2, 1).Range.Text = "3. Число, месяц, год и место рождения (село, деревня, город, район, область, край, республика, страна)";

oTable.Cell(3, 1).Range.Text = "4. Гражданство (если изменяли, то укажите, когда и по какой причине, если имеете гражданство другого государства укажите)";

oTable.Cell(4, 1).Range.Text = "5. Образование (когда и какие учебные заведения окончили, номера дипломов) Направление подготовки или специальность по диплому Квалификация по диплому";

oTable.Cell(5, 1).Range.Text = "6. Послевузовское профессиональное образование: адъюнктура, докторантура (наименование образовательного или научного учреждения, год окончания) Ученая степень, ученое звание (когда присвоены, номера дипломов, аттестатов)";

oTable.Cell(6, 1).Range.Text = "7. Какими иностранными языками и языками народов Российской Федерации владеете и в какой степени (читаете и переводите со словарем, читаете и можете объясняться, владеете свободно)";

oTable.Cell(1, 2).Range.Text = NameChange1.Text;

oTable.Cell(2, 2).Range.Text = BornInfo.Text;

oTable.Cell(3, 2).Range.Text = Civility.Text;

oTable.Cell(4, 2).Range.Text = Education.Text;

oTable.Cell(5, 2).Range.Text = PostUniversityEdu.Text;

oTable.Cell(6, 2).Range.Text = Languages.Text;

// Настройка границ таблицы

oTable.Borders[Word.WdBorderType.wdBorderTop].Visible = true;

oTable.Borders[Word.WdBorderType.wdBorderLeft].Visible = true;

oTable.Borders[Word.WdBorderType.wdBorderRight].Visible = true;

oTable.Borders[Word.WdBorderType.wdBorderBottom].Visible = true;

oTable.Borders[Word.WdBorderType.wdBorderVertical].Visible = true;

oTable.Borders[Word.WdBorderType.wdBorderHorizontal].Visible = true;

oDoc.SaveAs(path);

oWord.Visible = true;

oWord.Quit();

}

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

{

Surname.Clear();

UserName.Clear();

FatherName.Clear();

NameChange1.Clear();

BornInfo.Clear();

Civility.Clear();

PostUniversityEdu.Clear();

Languages.Clear();

}

}

}

## Графический пользовательский интерфейс программного обеспечения

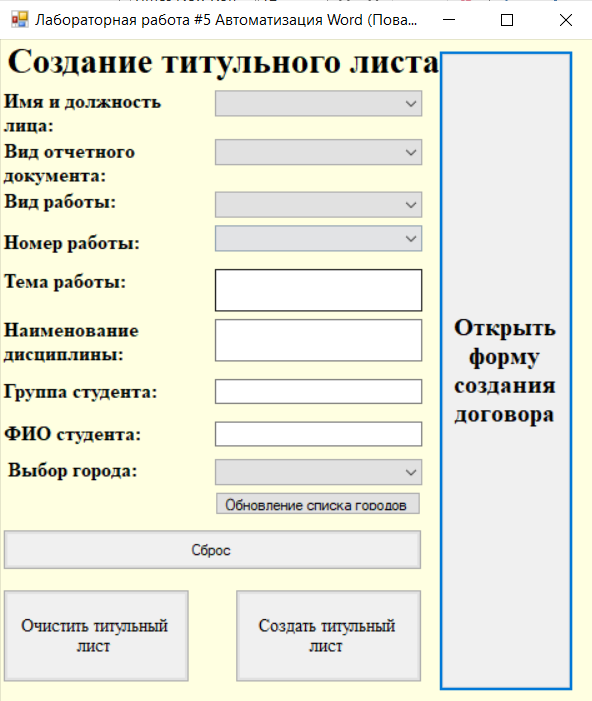


Рисунок 10 – Интерфейс основной формы

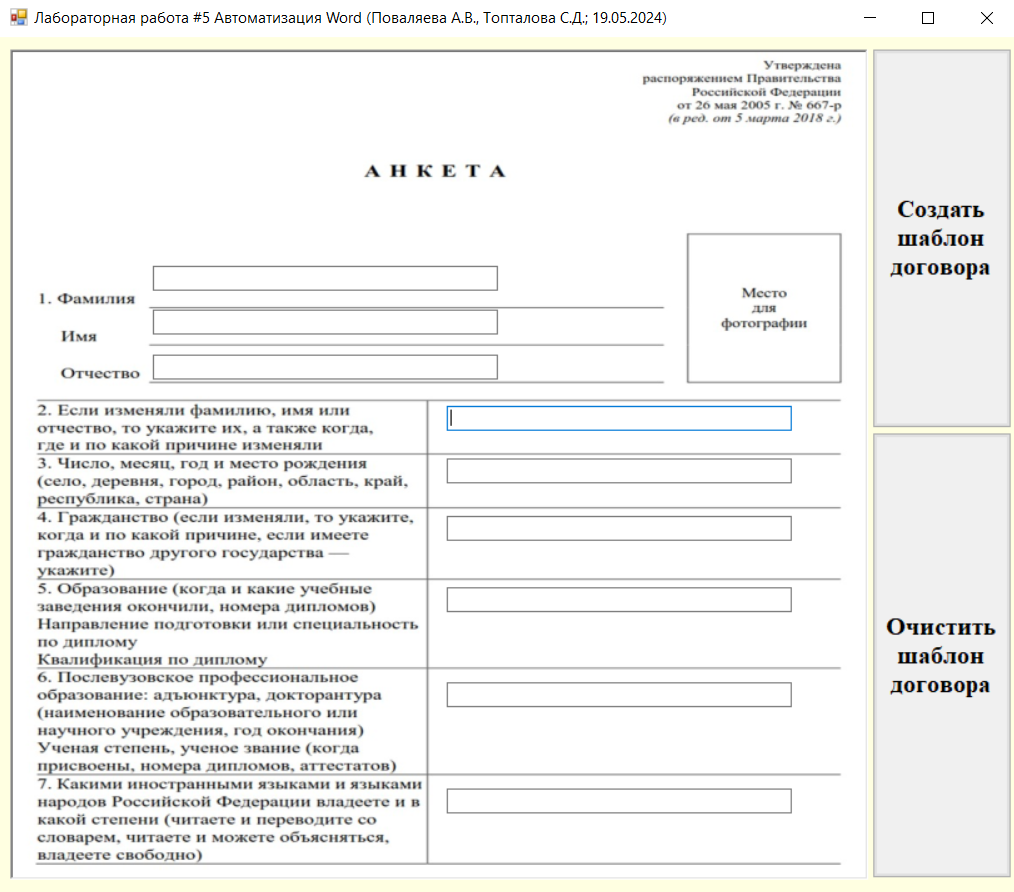


Рисунок 11 – Интерфейс формы анкеты

## Индивидуальное задание

2. Предоставить пользователю возможность выбора города, в котором выполнялась работа, из комбинированного списка. Перечень городов хранить в отдельном текстовом файле. Предусмотреть кнопку «Обновить» для загрузки города, введённого в перечень в процессе работы программы. Исключить необходимость перезапуска программы при добавлении/исключении города из перечня.

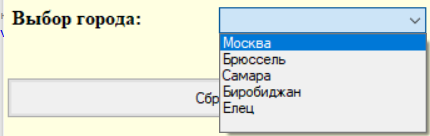


Рисунок 12 – Возможность выбора города

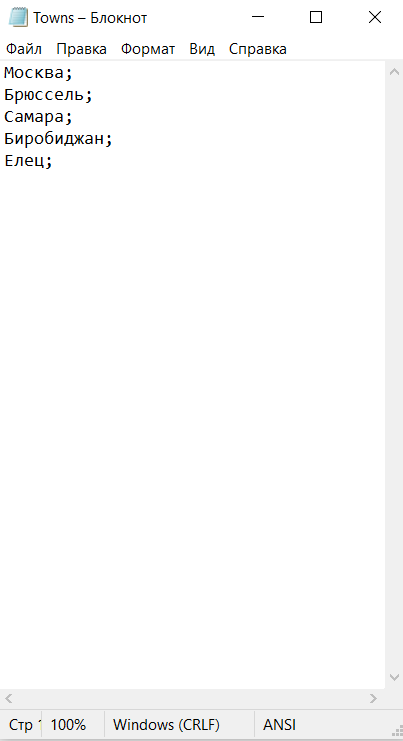


Рисунок 13 – Текстовый файл для хранения городов

## 

## Рисунок 14 – Кнопка обновления городов

## Расчёт тестовых примеров с использованием составленного программного обеспечения.

## 

Рисунок 14 – Результат первого тестового примера

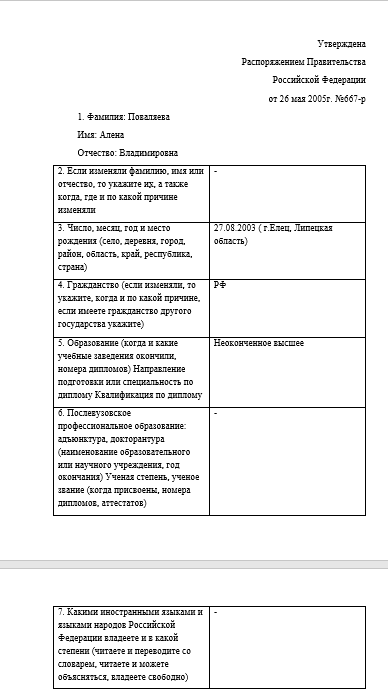


Рисунок 15 – Результат второго тестового примера

1. **Вывод по работе**

В текущей работе были освоены навыки подключения внешних модулей (библиотек классов) и использованы в разработанной программе.

Во время заполнения титульного листа реализовали запись по центру, по левому и по правому краю, применили полужирные начертание к тексу, а также изменяли размер шрифта конкретных предложений.  
 Создали и сохранили образец анкеты по варианту индивидуального задания, используя навыки программирования и библиотеку классов *Microsoft Office Word.*