

**Московский государственный технический  
университет им. Н.Э. Баумана.**

**Факультет «Информатика и системы управления»**

**Кафедра ИУ5. Курс «Базовые компоненты интернет технологий»**

**Отчет по лабораторной работе № 4**

**«Работа с файлами»**

Выполнил:

студент группы ИУ5-33

Терентьев Владислав

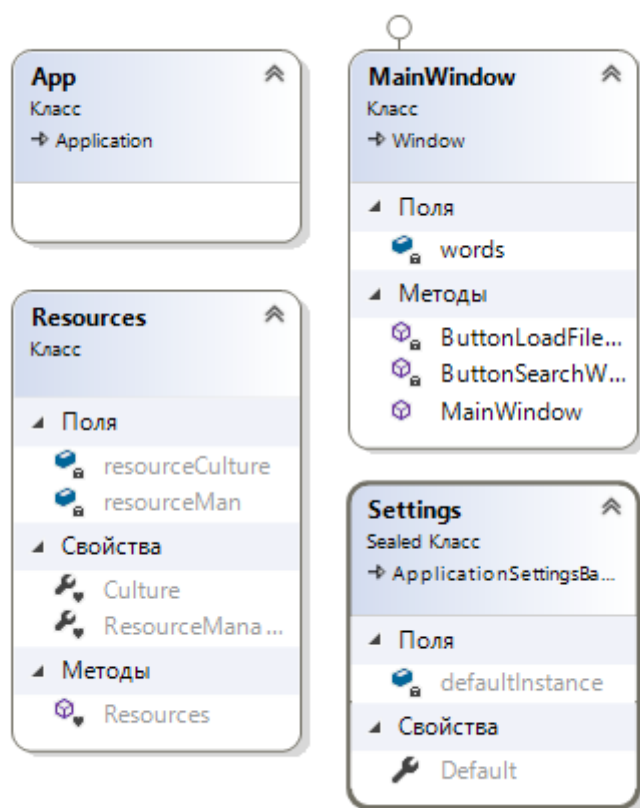
Москва, 2019 г.

## 1. Постановка задачи

Разработать программу, реализующую работу с файлами.

1. Программа должна быть разработана в виде приложения Windows Forms на языке C#. По желанию вместо Windows Forms возможно использование WPF (Windows Presentation Foundation).
2. Добавить кнопку, реализующую функцию чтения текстового файла в список слов `List<string>`.
3. Для выбора имени файла используется класс `OpenFileDialog`, который открывает диалоговое окно с выбором файла. Ограничить выбор только файлами с расширением «.txt».
4. Для чтения из файла рекомендуется использовать статический метод `ReadAllText()` класса `File` (пространство имен `System.IO`). Содержимое файла считывается методом `ReadAllText()` в виде одной строки, далее делится на слова с использованием метода `Split()` класса `string`. Слова сохраняются в список `List<string>`.
5. При сохранении слов в список `List<string>` дубликаты слов не записываются. Для проверки наличия слова в списке используется метод `Contains()`.
6. Вычислить время загрузки и сохранения в список с использованием класса `Stopwatch` (пространство имен `System.Diagnostics`). Вычисленное время вывести на форму в поле ввода (`TextBox`) или надпись (`Label`).
7. Добавить на форму поле ввода для поиска слова и кнопку поиска. При нажатии на кнопку поиска осуществлять поиск введенного слова в списке. Слово считается найденным, если оно входит в элемент списка как подстрока (метод `Contains()` класса `string`).
8. Добавить на форму список (`ListBox`). Найденные слова выводить в список с использованием метода «название\_списка.Items.Add()». Вызовы метода «название\_списка.Items.Add()» должны находиться между вызовами методов «название\_списка.BeginUpdate()» и «название\_списка.EndUpdate()».
9. Вычислить время поиска с использованием класса `Stopwatch`. Вычисленное время вывести на форму в поле ввода (`TextBox`) или надпись (`Label`).

## 2. Диаграмма классов



## 3. Текст программы

### 3.1. MainWindow.xaml

```
<Window x:Class="laba4_wpf.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        mc:Ignorable="d"
        Title="Терентьев Владислав ИУ5-33" Height="450" Width="650" MinHeight="220"
        MinWidth="560">
    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="25"></RowDefinition>
            <RowDefinition Height="25"></RowDefinition>
            <RowDefinition Height="35"></RowDefinition>
            <RowDefinition Height="50"></RowDefinition>
            <RowDefinition></RowDefinition>
        </Grid.RowDefinitions>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="200"></ColumnDefinition>
            <ColumnDefinition Width="Auto"></ColumnDefinition>
            <ColumnDefinition></ColumnDefinition>
        </Grid.ColumnDefinitions>
        <Button x:Name="ButtonLoadFile" Content="Чтение из файла" Grid.Column="0"
            Grid.Row="0" Grid.RowSpan="2" HorizontalAlignment="Center" VerticalAlignment="Center"
            Width="120" Height="30" Click="ButtonLoadFile_Click"/>
        <Label x:Name="LabelCountWords" Grid.Column="2" Grid.Row="1"
            HorizontalAlignment="Left" VerticalAlignment="Center"/>
```

```

        <TextBox x:Name="TextBoxSearchWord" TextWrapping="Wrap" Grid.Column="1"
Grid.Row="2" Grid.ColumnSpan="2" Width="215" Height="20" HorizontalAlignment="Left"
VerticalAlignment="Center"/>
        <Button x:Name="ButtonSearchWord" Content="Поиск" Grid.Column="0" Grid.Row="3"
HorizontalAlignment="Center" VerticalAlignment="Center" Width="120" Height="30"
Click="ButtonSearchWord_Click"/>
        <ListBox x:Name="ListBoxResult" Grid.Column="0" Grid.Row="4" Grid.ColumnSpan="3"
Margin="10"/>
        <Label x:Name="LabelTimeSearch" Grid.Column="2" Grid.Row="3"
HorizontalAlignment="Left" VerticalAlignment="Center"/>
        <Label x:Name="LabelTimer" Grid.Column="2" Grid.Row="0"
HorizontalAlignment="Left" VerticalAlignment="Center"/>
        <Label Content="Время чтения из файла: " Grid.Column="1" Grid.Row="0"
HorizontalAlignment="Center" VerticalAlignment="Center"/>
        <Label Content="Количество уникальных слов в файле: " Grid.Column="1"
Grid.Row="1" HorizontalAlignment="Center" VerticalAlignment="Center"/>
        <Label Content="Слово для поиска: " Grid.Column="0" Grid.Row="2"
HorizontalAlignment="Center" VerticalAlignment="Center"/>
        <Label Content="Время поиска: " Grid.Column="1" Grid.Row="3"
HorizontalAlignment="Center" VerticalAlignment="Center"/>
    </Grid>
</Window>

```

### 3.2. MainWindow.xaml.cs

```

using System.Collections.Generic;
using System.Windows;
using Microsoft.Win32;
using System.Diagnostics;
using System.IO;

namespace laba4_wpf
{
    /// <summary>
    /// Логика взаимодействия для MainWindow.xaml
    /// </summary>
    public partial class MainWindow : Window
    {
        public MainWindow()
        {
            InitializeComponent();

            readonly List<string> words = new List<string>();

            private void ButtonLoadFile_Click(object sender, RoutedEventArgs e)
            {
                OpenFileDialog fd = new OpenFileDialog
                {
                    Filter = "Текстовые Файлы|*.txt"
                };
                bool? result = fd.ShowDialog();
                if (result == true)
                {
                    Stopwatch timer = new Stopwatch();
                    timer.Start();
                    string text = File.ReadAllText(fd.FileName);
                    char[] separators = new char[] { ' ', '.', ',', '?', '!', '\\', '<', '>',
'/', '\\t', '\\n' };
                    string[] textArray = text.Split(separators);
                    foreach (string strTemp in textArray)
                    {
                        string str = strTemp.Trim();
                        if (str != "")

```

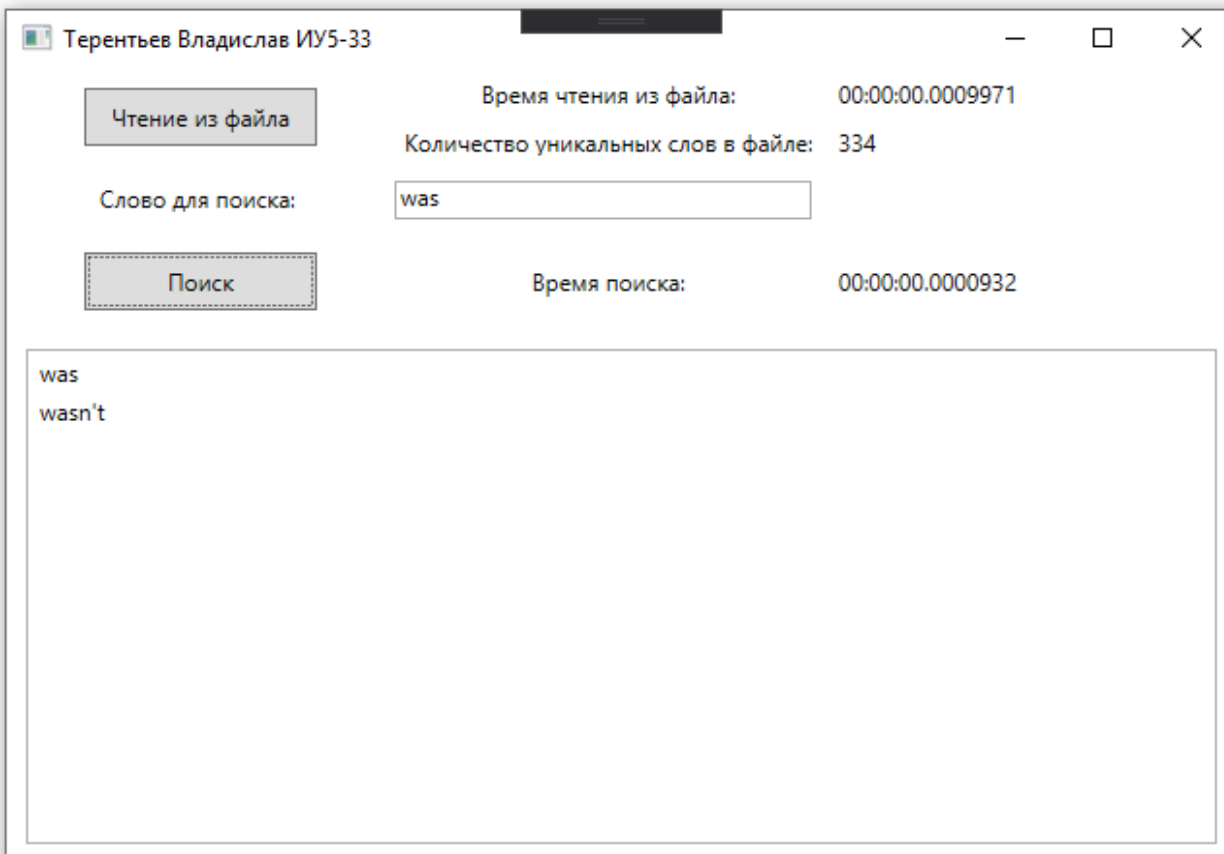
```

        {
            if (!words.Contains(str)) words.Add(str);
        }
    }
    timer.Stop();
    LabelTimer.Content = timer.Elapsed.ToString();
    LabelCountWords.Content = words.Count.ToString();
}
else
{
    MessageBox.Show("Необходимо выбрать файл");
}
}

private void ButtonSearchWord_Click(object sender, RoutedEventArgs e)
{
    string wordSearch = TextBoxSearchWord.Text.Trim();
    if (!string.IsNullOrEmpty(wordSearch) && words.Count > 0)
    {
        string wordUpper = wordSearch.ToUpper();
        List<string> tempList = new List<string>();
        Stopwatch timer = new Stopwatch();
        timer.Start();
        foreach (string str in words)
        {
            if (str.ToUpper().Contains(wordUpper))
            {
                tempList.Add(str);
            }
        }
        timer.Stop();
        LabelTimeSearch.Content = timer.Elapsed.ToString();
        ListBoxResult.ItemsSource=tempList;
    }
    else
    {
        MessageBox.Show("Выберите файл и введите слово для поиска");
    }
}
}
}
}
}

```

#### 4. Анализ результатов



Терентьев Владислав ИУ5-33

Чтение из файла

Слово для поиска: was

Поиск

Время чтения из файла: 00:00:00.0009971

Количество уникальных слов в файле: 334

Время поиска: 00:00:00.0000932

was  
wasn't

#### 5. Ссылка на репозиторий

<https://github.com/iYroglif/newlabs>