

Базовые компоненты интернет технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

Кочетков Михаил Дмитриевич
Группа ИУ5-316

10 октября 2018 г.

Задание

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
2. Добавить кнопку, реализующую функцию чтения 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()» должна находится 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`)

Код

Program.cs

```
1  using System;
2  using Eto.Forms;
3
4  namespace Lab_4.Desktop
5  {
6      class Program
7      {
8          [STAThread]
9          static void Main(string[] args)
10         {
11             new Application(Eto.Platform.Detect).Run(new MainForm());
12         }
13     }
14 }
```

MainForm.cs

```
1  using System.Collections.Generic;
2  using System.Diagnostics;
3  using System.IO;
4  using Eto.Drawing;
5  using Eto.Forms;
6
7  namespace Lab_4
8  {
9      public class MainForm : Form
10     {
11         public MainForm()
12         {
13             ClientSize = new Size(400, 400);
14             Title = "Lab 4";
15
16             var wordList = new List<string>();
17
18             var timeLabel = new Label();
19
20             var openFileButton = new Button { Text = "Open File" };
21             openFileButton.Click += delegate
22             {
23                 var openFileDialog = new OpenFileDialog
24                 {
25                     MultiSelect = false,
26                     Filters = {"Text|*.txt"}
27                 };
28
29                 var stopwatch = new Stopwatch();
30
31                 if (openFileDialog.ShowDialog(this) == DialogResult.Ok)
32                 {
33                     stopwatch.Start();
34                     var file = File.ReadAllText(openFileDialog.FileName);
35                     foreach (var word in file.Split(' '))
36                     {
37                         if (!wordList.Contains(word))
38                         {
39                             wordList.Add(word);
40                         }
41                     }
42                 }
43
44                 stopwatch.Stop();
45                 timeLabel.Text = "Time of opening and scanning: " +
46                     ↪ stopwatch.ElapsedMilliseconds + " ms";
47             };
48
49             var textBox = new TextBox();
50             var listBox = new ListBox();
```

```

50     var timeFindLabel = new Label();
51
52     var findButton = new Button { Text = "Find word" };
53     findButton.Click += delegate
54     {
55         listBox.Items.Clear();
56
57         var expectedSubstring = textBox.Text;
58         if (expectedSubstring.Trim(' ') == "")
59         {
60             listBox.Items.Add("Empty field");
61             return;
62         }
63
64         bool isFinded = false;
65
66         var stopWatch = new Stopwatch();
67         stopWatch.Start();
68
69         foreach (var word in wordList)
70         {
71             if (word.Contains(expectedSubstring))
72             {
73                 listBox.Items.Add(word);
74                 isFinded = true;
75             }
76         }
77
78         stopWatch.Stop();
79         if (!isFinded)
80         {
81             listBox.Items.Add("No matches");
82         }
83
84         timeFindLabel.Text = "Time of searching: " + stopWatch.ElapsedMilliseconds
85             ↪ + " ms";
86     };
87
88     var layout = new TableLayout
89     {
90         Padding = new Padding(10),
91         Spacing = new Size(5, 5),
92         Rows =
93         {
94             new TableRow(openFileButton, timeLabel),
95             new TableRow(textBox, findButton),
96             new TableRow(listBox, timeFindLabel)
97         }
98     };
99     Content = layout;
100 }

```

101 }
102 }

Тесты

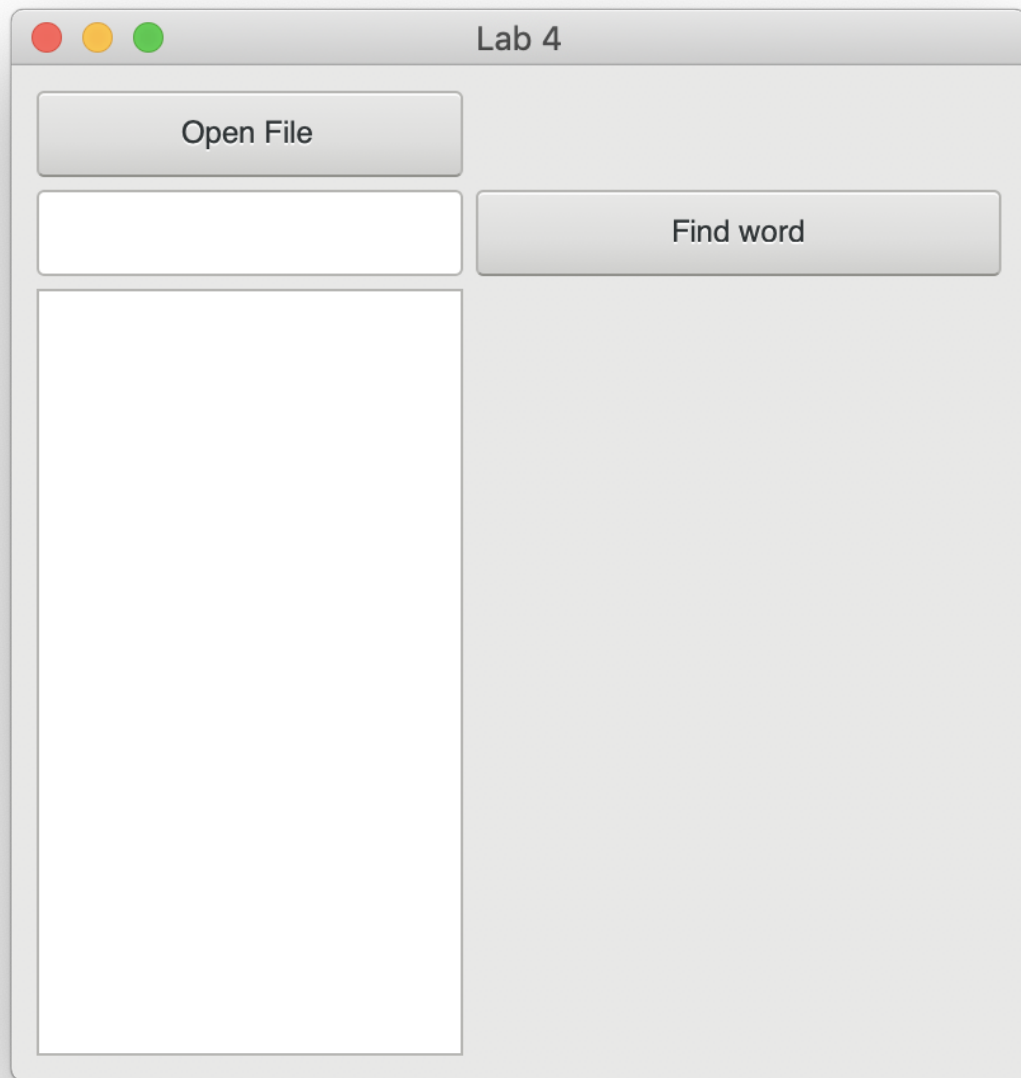


Рис. 1: Окно приложение после старте

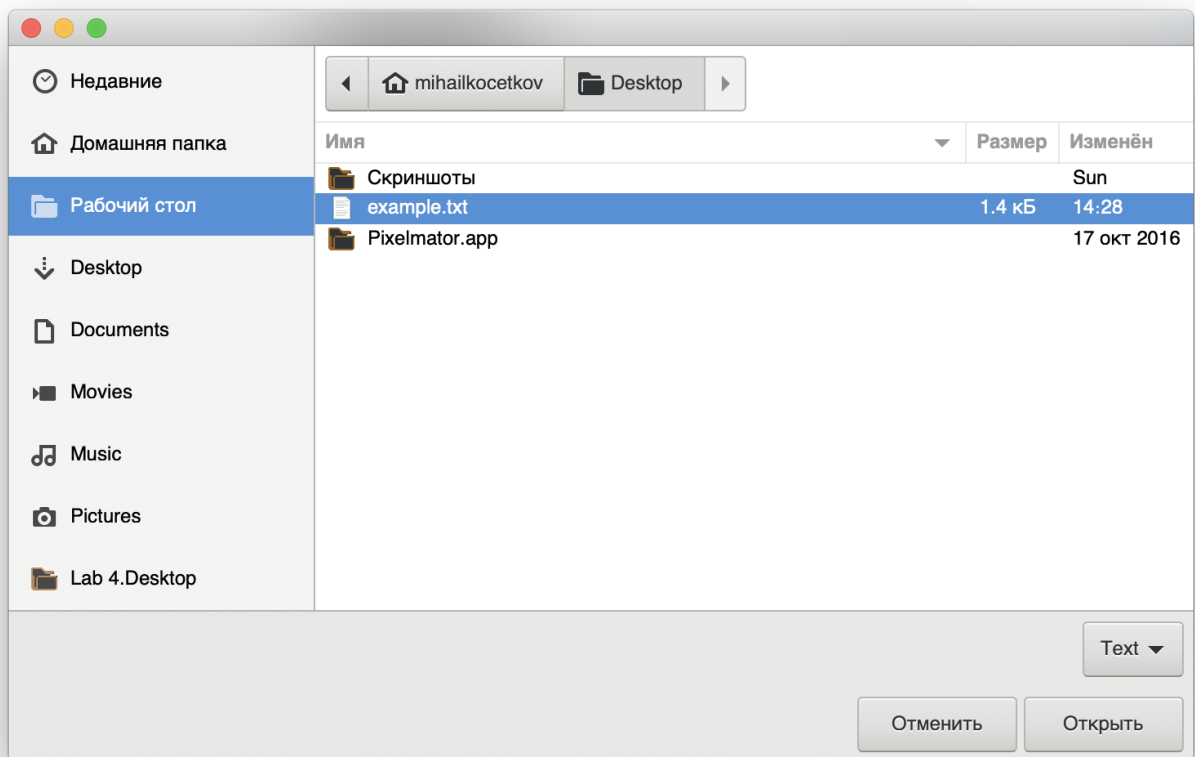


Рис. 2: Диалоговое окно открытия файла

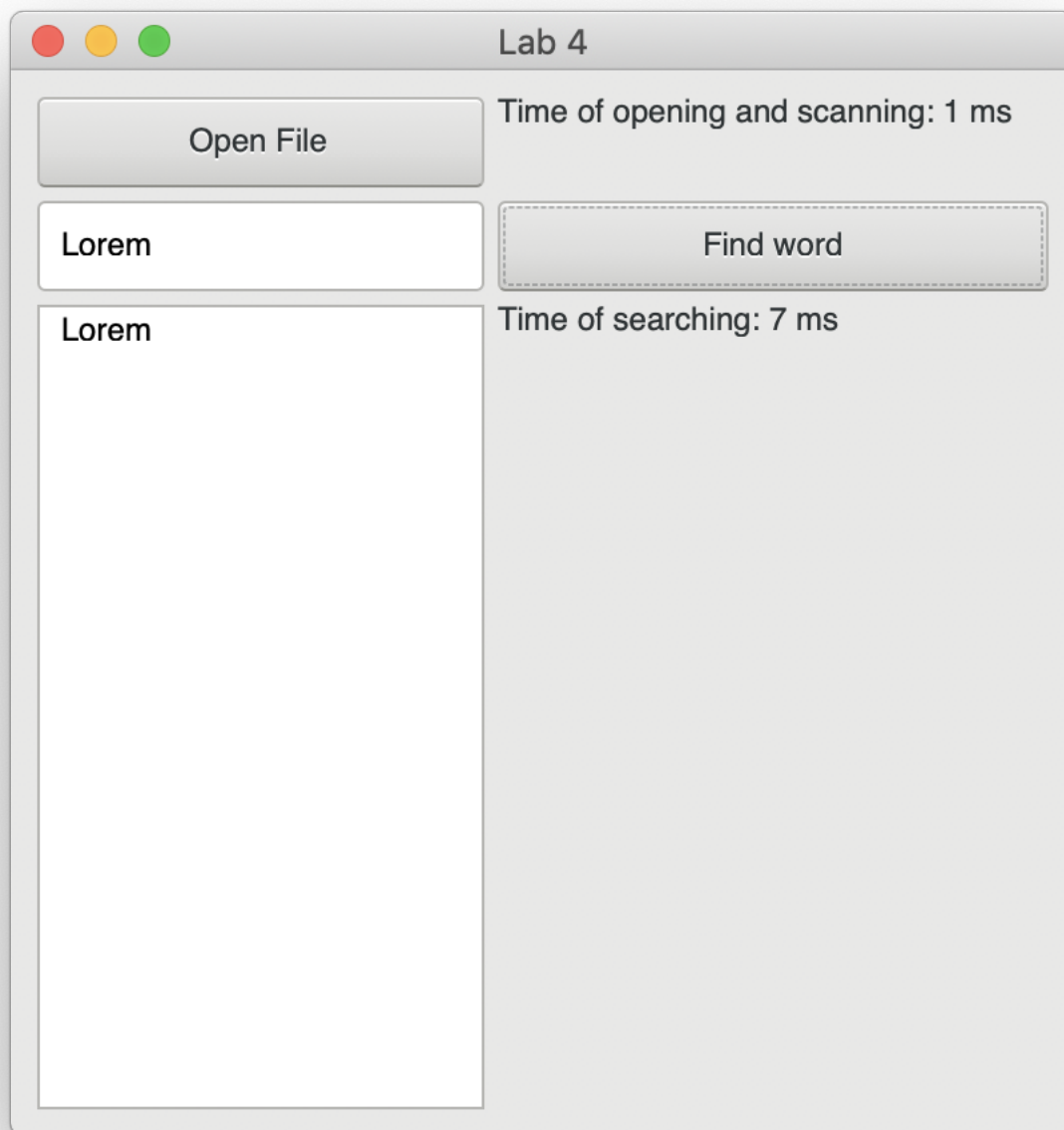


Рис. 3: Поиск целого слова

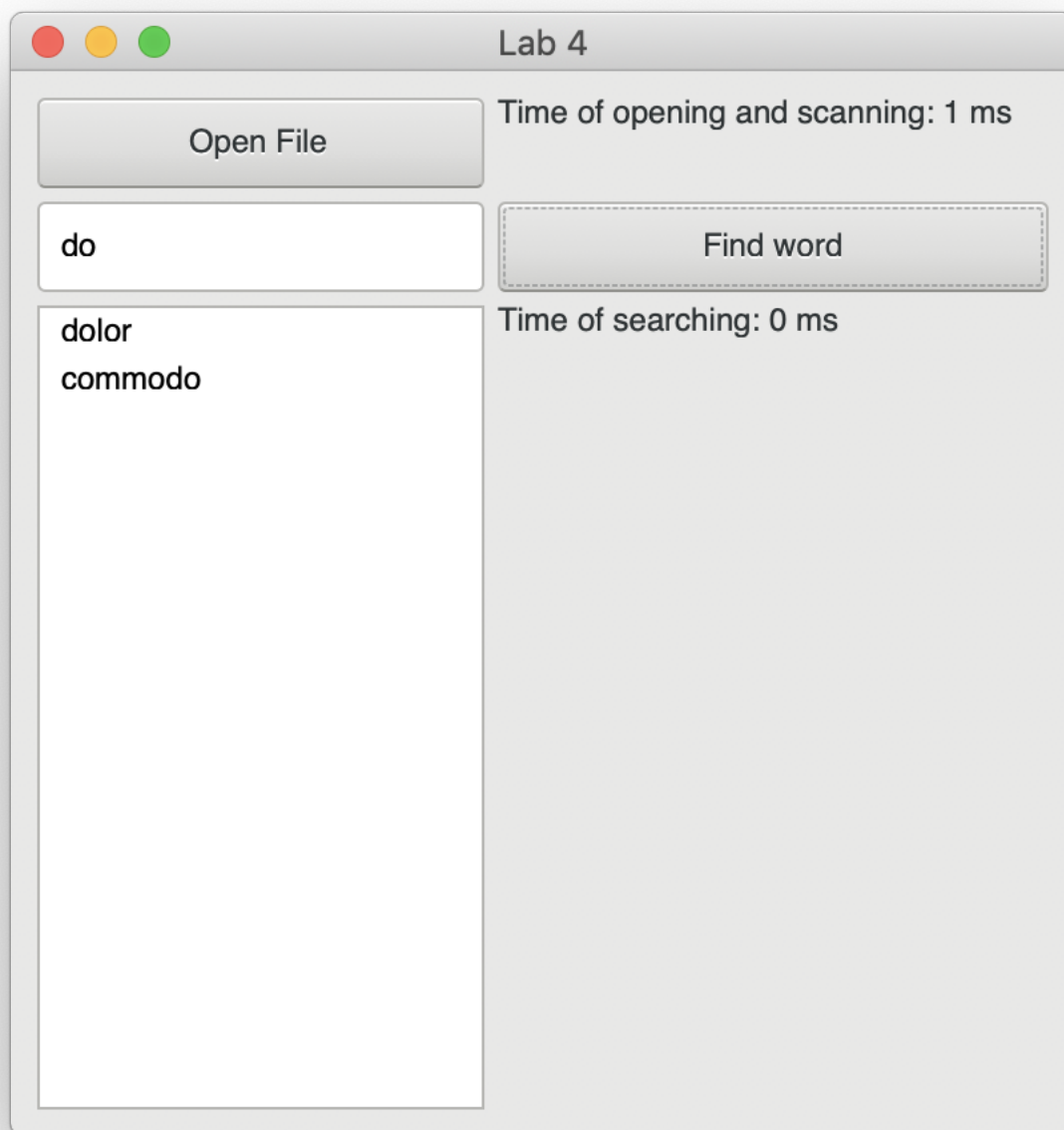


Рис. 4: Поиск слов по их части

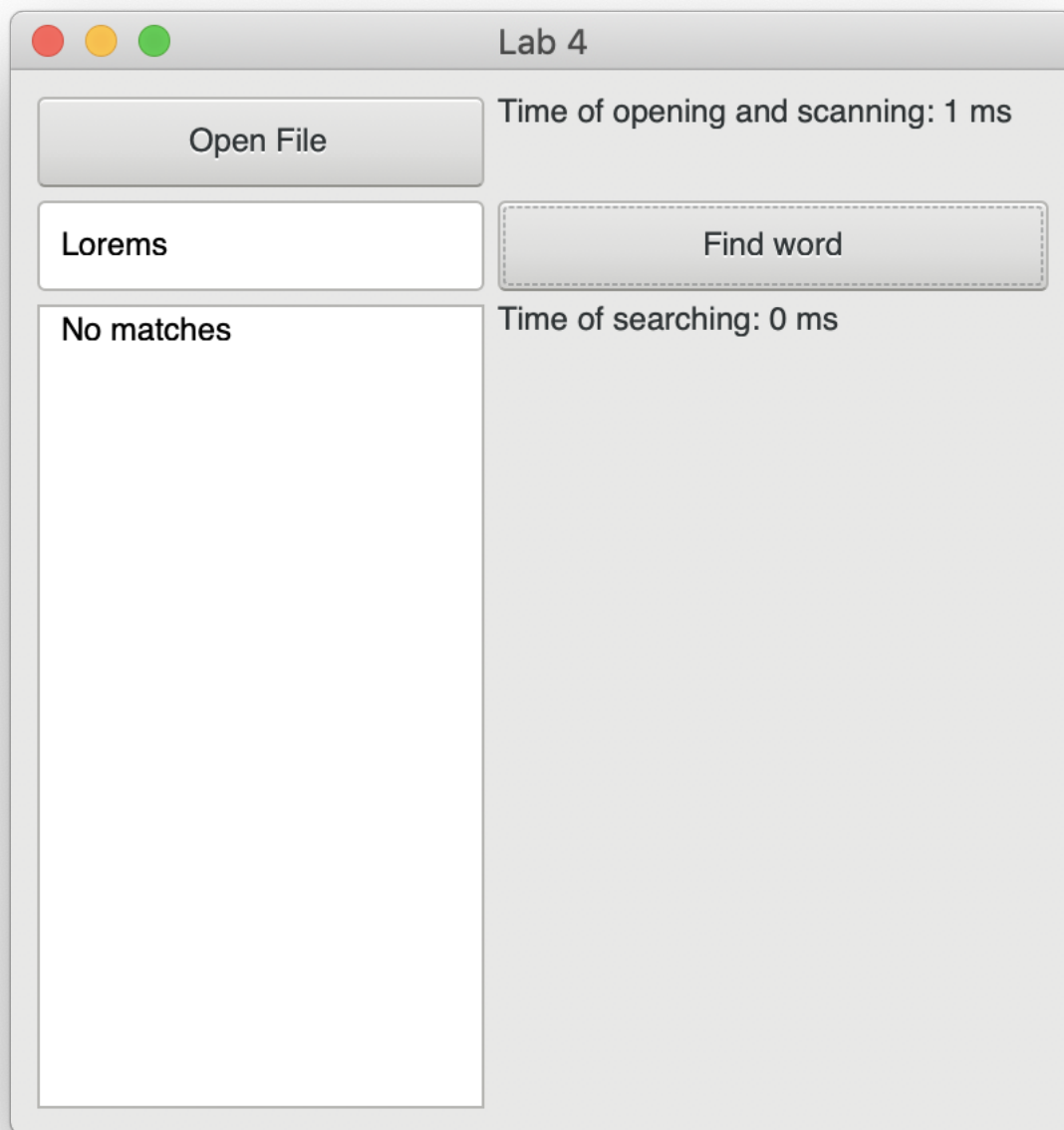


Рис. 5: Поиск несуществующего слова

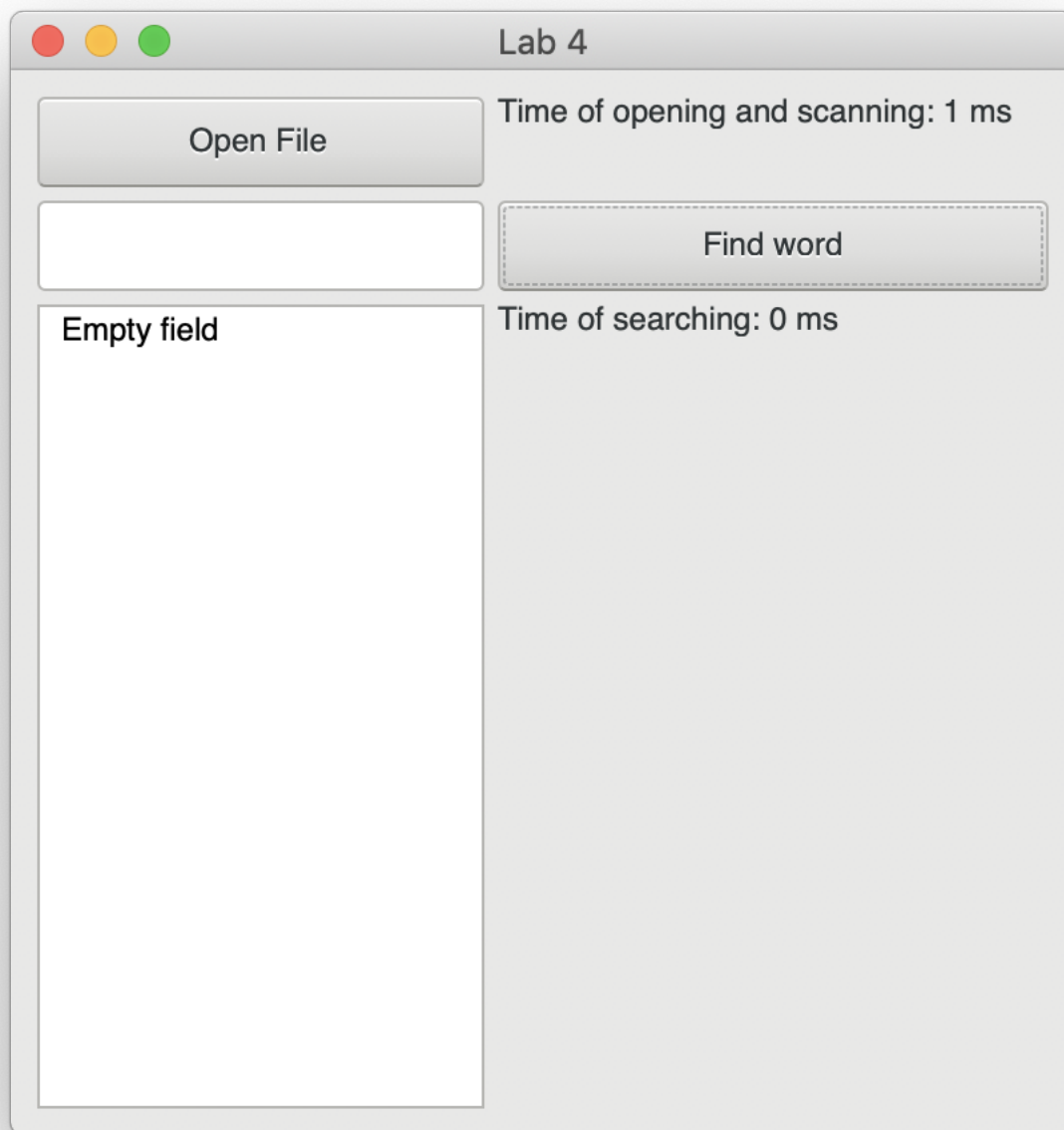


Рис. 6: Пустой ввод