

Лабораторная работа №2

Тема: Локализация WPF приложений.

Цели и задачи: Изучить основные принципы и методы разработки многоязычных приложений WPF.

Краткие сведения

Если вы планируете создавать многоязычное приложение, то на этапе проектирования и разработки, вам следует помнить несколько важных вещей. Даже при отсутствии таких планов можно ценой небольших усилий существенно облегчить их реализацию в последующих версиях приложения. Встроенные в платформу .NET Framework службы облегчают разработку одиночных приложений, которые могут быть адаптированы к различным языковым и региональным стандартам с помощью управляемой разработки с использованием Visual Studio.

Система работы с проектами Visual Studio предоставляет значительную поддержку для локализации приложений, работающих с формами Windows Forms. Существуют два способа создания файлов ресурсов при помощи среды разработки Visual Studio.

- Файлы ресурсов могут создаваться системой проектов для локализуемых элементов пользовательского интерфейса, таких как текст и изображения в форме. Затем файлы ресурсов будут встраиваться в сопутствующие сборки. Такие ресурсы называются ресурсами форм.

- Добавление шаблона файла ресурсов и затем редактирование шаблона с помощью конструктора XML. Второй способ применяется для создания локализуемых строк в

диалоговых окнах и сообщениях об ошибках. Затем необходимо написать код для доступа к этим ресурсам. Такие ресурсы называются ресурсами проекта.

В общем случае, ресурсы форм следует использовать для всех ресурсов, относящихся к форме в приложении. Ресурсы проекта следует использовать для всех строк и изображений, которые не основаны на форме, таких как сообщения об ошибках.

Пример создания приложения

1. Создадим проект WPF;
2. В дизайнере форм создадим простейший интерфейс входа в систему (рис. 1);

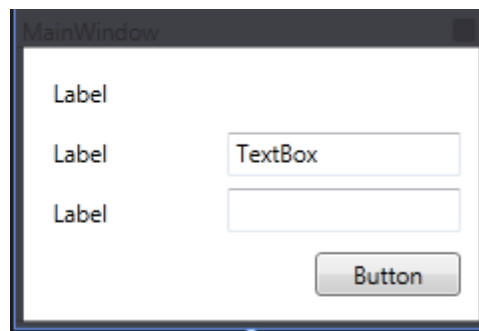


Рисунок 1 – Макет приложения

3. откроем автоматически созданный файл ресурсов WpfLocalized / Properties / Resources.resx и заполним строковые ресурсы необходимыми значениями, а также укажем модификатор доступа на «Public» (рис. 2);

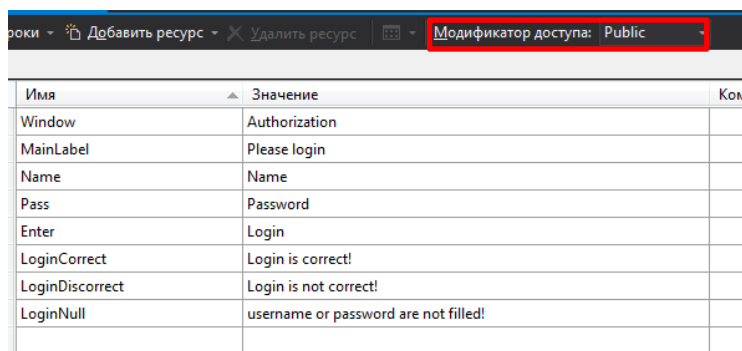
4. Добавим в код действия по кнопке;

```
private void bEnter_Click(object sender, RoutedEventArgs e)
{
    if (tLogin.Text.Equals("R00T") &&
        pPassword.Password.Equals("toor"))
    {
        MessageBox.Show(Properties.Resources.LoginCorrect,
            Properties.Resources.Window,
            MessageBoxButton.OK, MessageBoxImage.Information);
    }
}
```

```

    }
    else
    {
        MessageBox.Show(Properties.Resources.LoginDiscorrect,
            Properties.Resources.Window,
            MessageBoxButton.OK, MessageBoxImage.Error);
    }
}

```



Имя	Значение	Комм
Window	Authorization	
MainLabel	Please login	
Name	Name	
Pass	Password	
Enter	Login	
LoginCorrect	Login is correct!	
LoginDiscorrect	Login is not correct!	
LoginNull	username or password are not filled!	

Рисунок 2 – Строковые параметры

5. Объявим собственное пространство имен для XAML;
`xmlns:res="clr-namespace:WpfLocalized.Properties"`

6. Создадим привязки элементов к созданным строковым ресурсам (прописывать необходимо вручную);

`<Label Content="{x:Static res:Resources.MainLabel}" Margin="10,10,10,0" VerticalAlignment="Top" FontStyle="Italic" HorizontalAlignment="Center"/>`

7. Создадим русскую локализацию. Для этого скопируем файл ресурсов в новый `Resources.{cl-LN}.resx`, где `cl` – культура (страна), а `LN` – язык. Например, для русского языка в России файл ресурсов будет называться `Resources.ru-RU.resx`. и переведем надписи на русский язык. Чтобы избежать лишних файлов в проекте необходимо поставить для этого ресурса модификатор доступа «Без создания кода» (рис. 3);

Добавить ресурс × Удалить ресурс Модификатор доступа: Без создания		
Имя	Значение	Комме
Enter	Вход	
LoginCorrect	Вход выполнен успешно	
LoginDiscorrect	Имя пользователя или пароль не верны	
LoginNull	Имя пользователя или пароль не заполнены	
MainLabel	Пожалуйста авторизуйтесь	
Name	Имя пользователя	
Pass	Пароль	
Window	Авторизация	

Рисунок 3 – Русская локализация

8. При запуске программы среда выполнения автоматически определяет страну и язык установленные в системе и выбирает необходимую локализацию (рис. 4);

9. Для принудительного вызова англоязычной локализации добавим в конструктор, до вызова метода `InitializeComponent`, следующую строчку и запустим приложение (рис. 5);

```
System.Threading.Thread.CurrentThread.CurrentUICulture = System.Globalization.CultureInfo.GetCultureInfo("en-US");
```

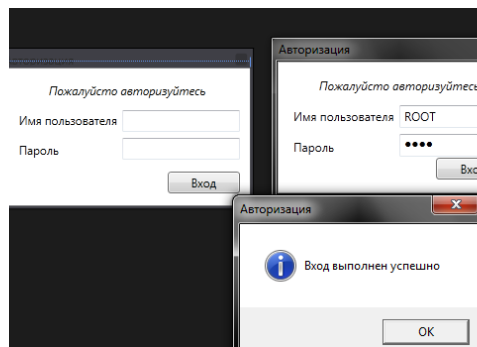


Рисунок 4 – Тестирование приложения

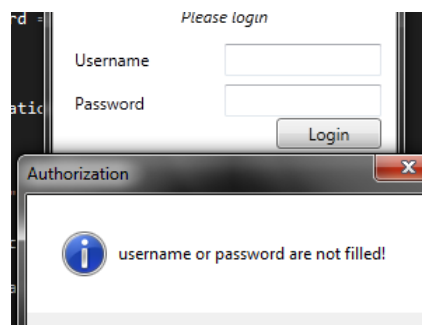


Рисунок 5 –Тестирование английской локализации

Итоговая XAML разметка для тестового приложения

```
<Window x:Class="WpfLocalized.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:res="clr-namespace:WpfLocalized.Properties"
        Title="{x:Static res:Resources.Window}"
        Height="164.766"
        Width="259.941"
        WindowStartupLocation="CenterScreen"
        WindowStyle="ToolWindow"
        ResizeMode="NoResize">
    <Grid>
        <Label Content="{x:Static res:Resources.MainLabel}"
              Margin="10,10,10,0"
              VerticalAlignment="Top"
              FontStyle="Italic"
              HorizontalAlignment="Center"/>
        <Label Content="{x:Static res:Resources.Name}"
              HorizontalAlignment="Left"
              Margin="10,41,0,0"
              VerticalAlignment="Top"/>
        <Label Content="{x:Static res:Resources.Pass}"
              HorizontalAlignment="Left"
              Margin="10,72,0,0"
              VerticalAlignment="Top"/>
        <TextBox x:Name="tLogin"
              Height="23" Margin="122,43,10,0"
              TextWrapping="Wrap"
              VerticalAlignment="Top"/>
        <PasswordBox x:Name="pPassword"
              Margin="122,72,10,0"
              VerticalAlignment="Top"
              Height="23"/>
        <Button x:Name="bEnter"
              Content="{x:Static res:Resources.Enter}"
              Margin="0,0,10,13"
              Height="22" VerticalAlignment="Bottom"
              Grid.RowSpan="2"
              HorizontalAlignment="Right"
              Width="75"
              Click="bEnter_Click"/>
    </Grid>
</Window>
```

Задания на лабораторную работу

1. Изучить материал, представленный в данных методических указаниях к лабораторной работе;
2. Разработать WPF приложение, согласно заданию преподавателя;
3. Составить отчет. В отчете отобразить:
 - цели и задачи лабораторной работы;
 - XAML разметку;

- исходный код приложения (отображающий только логику программы и описание файлов ресурсов);
- скриншоты работы приложения;
- выводы по проделанной лабораторной работе.