

Практическая работа №19

Настройка интерфейса приложения

1 Цель работы

- 1.1 Изучить основные элементы графического интерфейса Avalonia UI.

2 Литература

- 2.1 Avalonia documentation – Текст : электронный // AvaloniaUI, 2024. – URL: <https://docs.avaloniaui.net/>

3 Подготовка к работе

- 3.1 Повторить теоретический материал (см.п.2).
3.2 Изучить описание лабораторной работы.

4 Основное оборудование

- 4.1 Персональный компьютер.

5 Задание

- 5.1 Создайте новый проект Avalonia

5.1.1 Создать новый проект типа Avalonia C# Project и назвать его HelloApp

При создании:

- выбрать платформу Desktop, Android,
- указать фреймворк CommunityToolkit

5.1.2 Изменить запускаемый проект на HelloApp.Desktop. Запустить приложение.

5.1.3 Изменить запускаемый проект на HelloApp.Android. Запустить приложение.

- 5.2 Добавление страниц

5.2.1 Добавьте в приложение новую страницу HomeView (Добавить – Создать новый элемент – Avalonia – Avalonia User Control)

5.2.2 Измените пространство имен HomeView на HelloApp.Views (измените .axaml и .axaml.cs файлы)

- 5.2.3 На странице расположите TextBlock с текстом «Hello Avalonia!»

5.2.4 В MainWindow.axaml измените запускаемую страницу на HomeView. Запустите приложение.

5.2.5 Измените параметры TextBlock. Выровняйте его по центру экрана, измените цвет текста на зеленый, размер шрифта – 40

- 5.3 Взаимодействие разметки с кодом

5.3.1 В HomeView добавьте StackPanel и переместите TextBlock в нее.

5.3.2 Добавьте в StackPanel кнопку «Изменить текст»

5.3.3 Создать в папке ViewModels новый класс HomeViewModel. Данный класс должен быть наследником класса ViewModelBase и иметь модификатор

partial.

5.3.4 В классе определить поле с атрибутом `ObservableProperty` и метод с атрибутом `RelayCommand`:

```
[ObservableProperty]
private string text;

[RelayCommand]
public void ChangeText() {
    Text = "НОВЫЙ текст"
}
```

5.3.5 Изучить описание данных атрибутов. Пересобрать проект

5.3.6 Откройте файл `HomeView.axaml.cs` и в конструкторе установите значение `DataContext`:

```
public HomeView()
{
    InitializeComponent();
    DataContext = new HomeViewModel();
}
```

5.3.7 В разметку `UserControl` страницы добавьте строки

```
xmlns:vm="using:HelloApp.ViewModels"
x:DataType="vm:HomeViewModel"
```

5.3.8 На странице привяжите значение `Text` элемента `TextBlock` к соответствующему свойству `HomeViewModel` и значение `Command` элемента `Button` к команде `ChangeTextCommand`.

```
<TextBlock Text="{Binding Text}" ... />
<Button Command="{Binding ChangeTextCommand}" ... />
```

5.4 Контейнеры элементов управления

5.4.1 Добавить на экран `DockPanel`, задать для нее размеры.

5.4.2 В панели расположить 5 `Rectangle` разных цветов. Поочередно для них указать свойства:

- `DockPanel.Dock="Top" Height="100"`
- `DockPanel.Dock="Left" Width="100"`
- `DockPanel.Dock="Bottom" Height="100"`
- `DockPanel.Dock="Right" Width="100"`

Для последнего `Rectangle` свойство не устанавливайте.

5.4.3 Добавить на экран `Grid`, задать для него размеры и параметры `Grid.ColumnDefinitions="*,2*"`, `Grid.RowDefinitions="Auto,100"`

5.4.4 В сетке расположить 4 `Rectangle` разных цветов, расположив их в разных ячейках `Grid` при помощи `Grid.Column` и `Grid.Row`

5.4.5 Добавить на экран две `StackPanel`, для одной указать `Orientation="Horizontal"`, для другой `Orientation="Vertical"`.

5.4.6 В панелях расположить по 4 `TextBlock` с текстом, указать им различный размер.

6 Порядок выполнения работы

- 6.1 Выполнить все задания из п.5.
- 6.2 Ответить на контрольные вопросы.

7 Содержание отчета

- 7.1 Титульный лист
- 7.2 Цель работы
- 7.3 Ответы на контрольные вопросы
- 7.4 Вывод

8 Контрольные вопросы

- 8.1 Что такое AXAML?
- 8.2 Что такое Avalonia UI?
- 8.3 Как создать проект Avalonia UI?
- 8.4 Какие контейнеры компоновки используются в приложениях Avalonia UI