

Лабораторная работа №26

Разработка интерфейса пользователя: компоновка элементов

1 Цель работы

1.1 Изучить элементы-контейнеры, применяющиеся в приложениях WPF для компоновки.

2 Литература

2.1 <https://metanit.com/sharp/wpf/> – гл.3.

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

При разработке WPF-приложений используется XAML – язык разметки, используемый для инициализации объектов в технологиях на платформе .NET.

Синтаксис языка XAML похож на HTML и XML.

В тэгах описываются элементы управления, в атрибутах – свойства элементов управления.

Window (окно) – аналог тэга body в HTML – внутри него размещаются все остальные элементы управления.

5.1 Применение контейнера StackPanel для создания меню переходов

5.1.1 Создать приложение WPF.

5.1.2 В открывшемся окне, отображающем внешний вид окна (Конструктор) и разметку (XAML) вместо элемента Grid написать StackPanel (закрывающийся тэг изменится автоматически).

5.1.3 Разместить в StackPanel кнопки со следующими подписями (Content):

- Задание 1,
- Задание 2,
- Задание 3,
- Задание 4,
- Задание 5.

5.1.4 У каждой кнопки указать ширину 200, а у StackPanel – выравнивание по горизонтали (HorizontalAlignment) и выравнивание по вертикали (VerticalAlignment) – по центру.

5.1.5 Добавить в приложение набор окон (названия оставить по умолчанию). Добавить каждой кнопке имя (TaskNButton, вместо N – номер задания). При нажатии на каждую кнопку реализовать переход к соответствующему окну (окна будут использоваться в следующих заданиях).

5.1.6 В окне Window1 скопировать StackPanel из окна MainWindow. Сменить у StackPanel ориентацию (Orientation) на горизонтальную, выравнивание по горизонтали убрать, по вертикали – по верхнему краю. Сравнить результат с предыдущим (в п.5.1.4).

5.2 Применение контейнера Grid (сетки) для создания формы авторизации

5.2.1 В окне Window2 изменить содержимое элемента Grid:

- поставить курсор между тэгами **<Grid>** и **</Grid>**,
- во вкладке Свойства в разделе Макет нажать на «...» у **ColumnDefinitions** (определение столбцов) и добавить 2 столбца,
- во вкладке Свойства в разделе Макет нажать на «...» у **RowDefinitions** (определение строк) и добавить 3 строки.

5.2.2 Разместить на сетке элементы управления:

- перенести две метки с текстом Логин и Пароль в ячейки левого столбца сетки (текст меток меняется в свойстве **Content**)
- перенести поля ввода для указания логина и пароля в ячейки правого столбца сетки
- перенести кнопку с подписью «Авторизоваться» в нижнюю строку, последний столбец.

5.2.3 Выровнять элементы управления по сетке:

- убрать из тэгов **Label**, **TextBox**, **Button** все свойства кроме **Content**
- используя вкладку Свойства или разметку XAML, указать, в каких строках (**Grid.Row**) и столбцах (**Grid.Column**) находятся элементы (нумерация строк и столбцов с нуля)

5.2.4 Автоматическое вычисление ширины и высоты ячеек

- у все тэгов **RowDefinition** указать **Height="Auto"** для того, чтобы высота строк соответствовала высоте элементов управления в строках
- у левого столбца внутри тэга **ColumnDefinition** указать **Width="Auto"** для того, чтобы ширина столбца соответствовала ширине элементов управления в столбцах

5.3 Применение контейнера GridSplitter (две панели с разделителем) для создания интерфейса, аналогичного проводнику

5.3.1 В окне Window3 изменить содержимое элемента Grid:

- поставить курсор между тэгами **<Grid>** и **</Grid>**,
- добавить в сетку три столбца, ширина центрального столбца – **Auto**, ширина левого – **0.5***, ширина правого – ***** (т.е., ширина пропорциональна и у левого – 50% от ширины правого),
- добавить в сетку две строки, высота верхней – 40 пикселей.

5.3.2 Разместить на сетке элементы управления:

В верхнюю строку перенести элемент **StackPanel** (указать, что занимает 3 столбца (**Grid.ColumnSpan**)),

В нижнюю строку перенести:

- в левый столбец элемент **TreeView**,
- в правый столбец элемент **Listview**,
- в центральный столбец элемент **GridSplitter**.

5.3.3 У GridSplitter указать следующие настройки:

- выравнивание по горизонтали: по центру,
- выравнивание по вертикали: растянуть (**Stretch**),
- ширина – 3.

В **StackPanel** горизонтально разместить надпись «Поиск» и поле ввода.

5.4 Применение контейнера DockPanel (панели с привязкой к краям окна) для создания интерфейса, аналогичного разметке сайта с шапкой, подвалом, боковым меню.

5.4.1 В окне Window4 вместо элемента Grid написать DockPanel (закрывающийся тэг изменится автоматически).

5.4.2 Разместить в DockPanel метки со следующими подписями:

- Шапка,
- Меню,
- Контент,
- Подвал.

Цвет меток (Background) сделать различным.

Метка «Шапка» должна быть выровнена по верхнему краю, «Подвал» - по нижнему, «Меню» - в области между шапкой и подвалом с выравниванием по левому краю, «Контент» - вся оставшаяся область.

5.5 Применение контейнера WrapPanel для отображения меню.

5.5.1 В окне Window5 скопировать разметку из приложения 5.4

5.5.2 Вместо метки «Шапка» разместить WrapPanel с кнопками, совпадающими с кнопками из п.5.1.6.

5.5.3 Проверить, как отображаются элементы в WrapPanel при уменьшении ширины окна.

5.6 Применение контейнера Canvas для позиционирования относительно краев.

5.6.1 В окне Window5 вместо метки Контент разместить Canvas с четырьмя объектами Image. Каждое изображение разместить в своем углу (левый-верхний, левый-нижний, правый-верхний, правый-нижний), у каждого изображения задать Source (картинку)

5.6.2 У картинок, для того, чтобы они оставались в исходном углу, настроить привязки к требуемым краям (Left/Right/Top/Bottom), используя атрибут Canvas.Край="отступ"

Например: Canvas.Right="30" (отступ справа 30 пикселей)

5.6.3 Проверить, как отображаются элементы в Canvas при изменении размера окна.

6Порядок выполнения работы

6.1 Выполнить все задания из п.3 в одном WPF-проекте LabWork26.

6.2 Ответить на контрольные вопросы.

7Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8Контрольные вопросы

8.1 Что такое «компоновка» в WPF?

8.2 Какой класс является родительским для всех элементов-контейнеров в WPF?

8.3 Как выравнивать элементы внутри контейнера по высоте и по ширине?

8.4 В чем особенность компоновки с использованием следующих элементов-контейнеров: Grid и Canvas?

8.5 Чем отличается компоновка с использованием StackPanel, DockPanel, WrapPanel?

8.6 В каких единицах измерения могут задаваться размеры элементов в приложениях WPF?