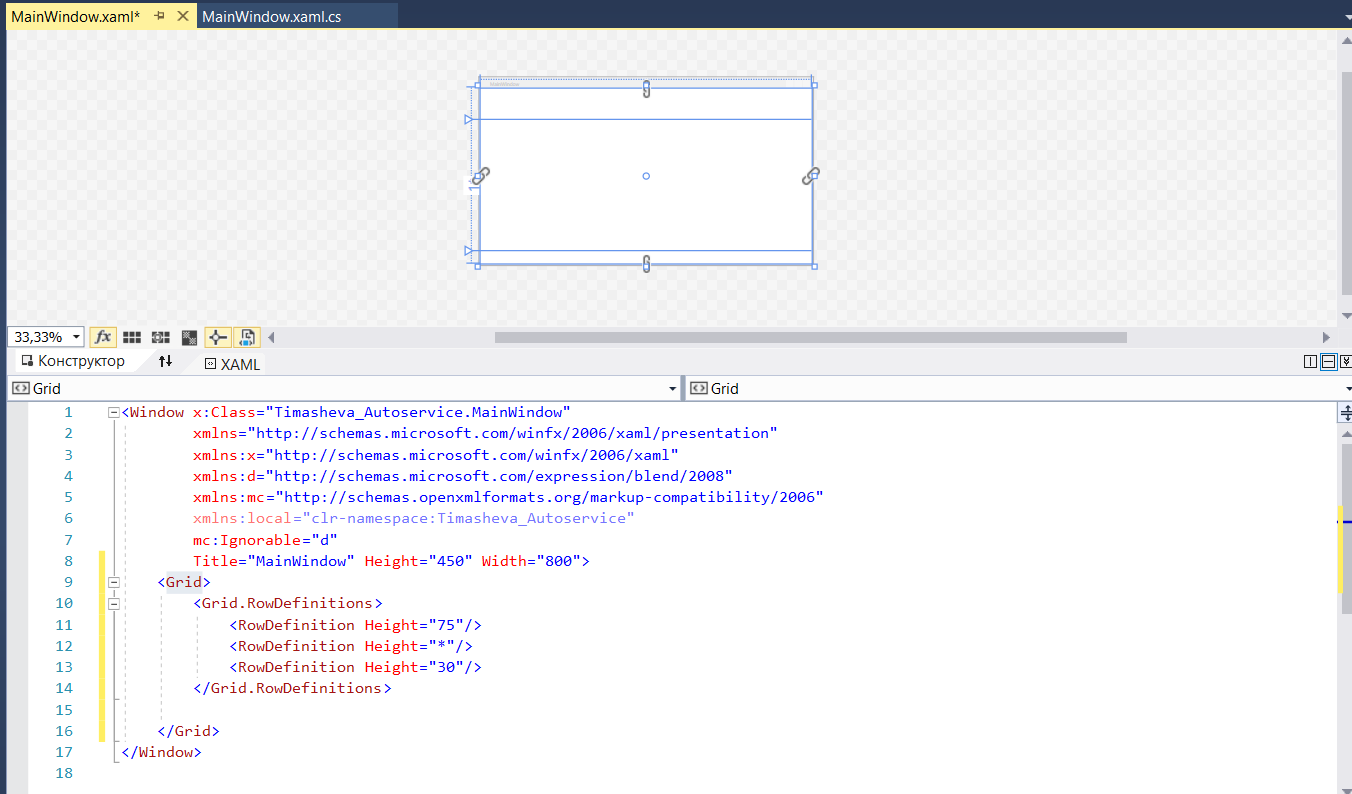
Создание каркаса приложения Familiya\_Autoservice

1. Запустить VS. Создать проект WPF, назвать Familiya\_Autoservice.

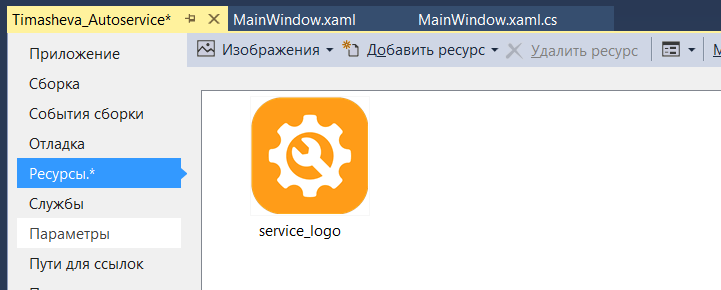
2. Задать следующие ряды:



3. Добавим картинку логотипа в ресурсы приложения. Для этого в Обозревателе решений (панель справа) открываем Properties, выбираем вкладку Ресурсы:

Поменять тип ресурса со Строки на Изображения.

Найти через Проводник рисунок логотипа service\_logo.png и перетащить его в рабочую область:

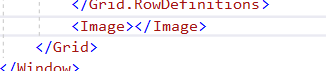


Тогда в Обозревателе решений появляется папка Resources с содержимым:

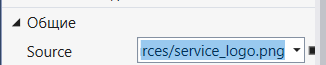
Выбрать картинку, и для нее поменять свойство Действие при сборке, поставив ему значение Resource:

4. Через Обозреватель решений вернуться в окно MainWindow.xaml.

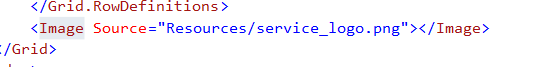
Добавить на форму Image:



В свойствах его указать:

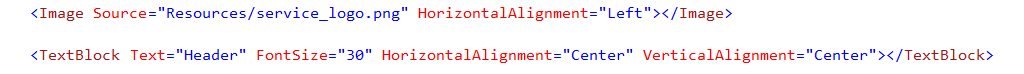


Тогда автоматически пропишется:

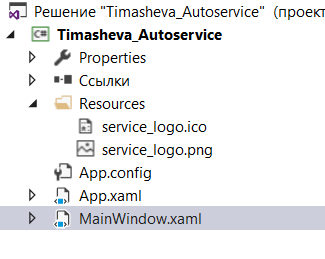


Добавим еще свойство для картинки: HorizontalAlignment="Left"

5. Добавим после рисунка компонент textblock:



В Обозревателе решений – Вкладка Ресурсы (Rеsources): добавить иконку путем перетаскивания:



В Title главной формы добавим свойства семейство шрифтов (взято из Руководства по стилю) и иконку:



Для установки иконки прописать имя файла ico.

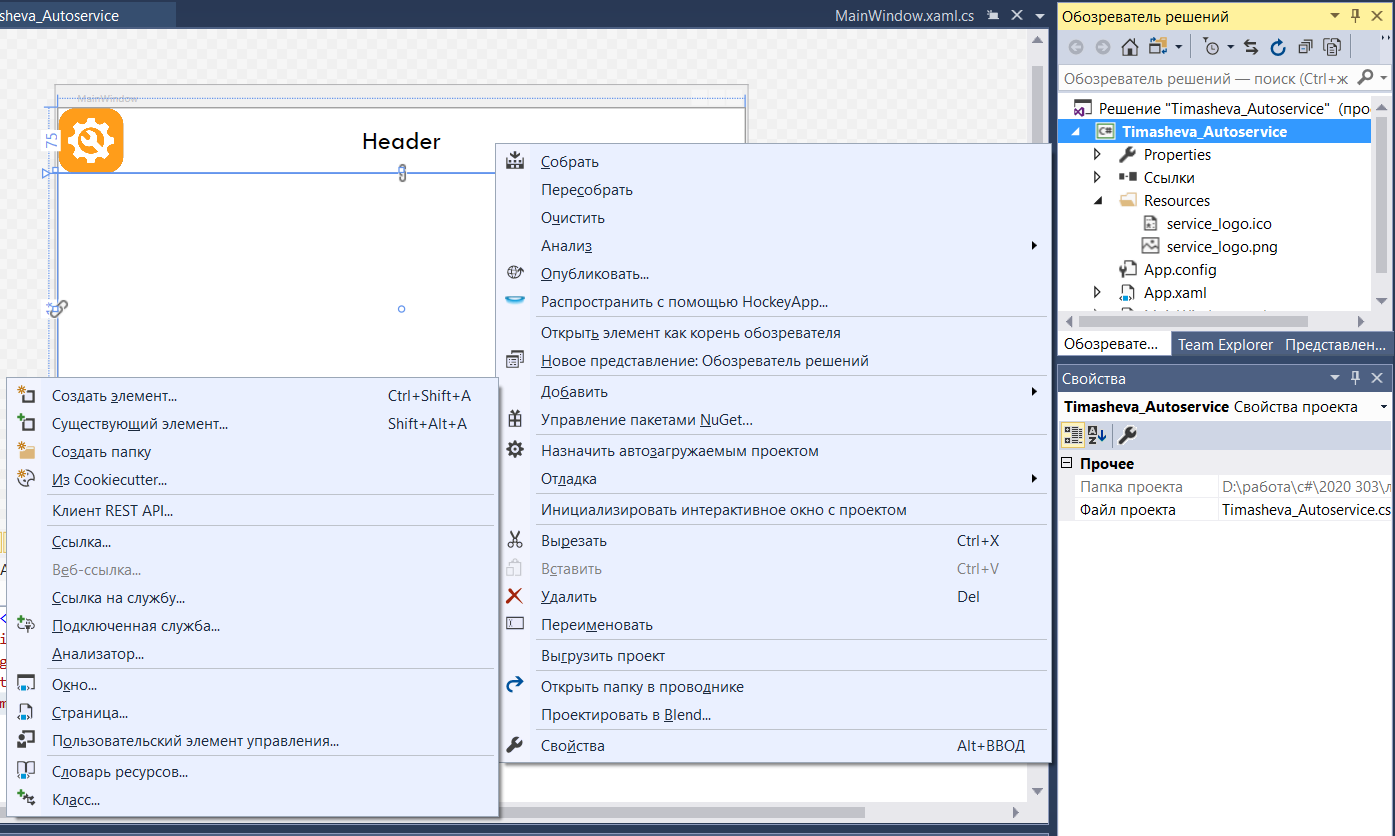
Если не удается поставить именно файл ico в качестве иконки, можно поставить файл png:



6. Определим в центральную область фрейм:



7. Добавим страницу для просмотра сервисов (ServicePage): правой кнопкой по названию решения (проекта) в Обозревателе решений, затем пункт Добавить 🡪 Страница:



Дадим ей имя ServicePage:

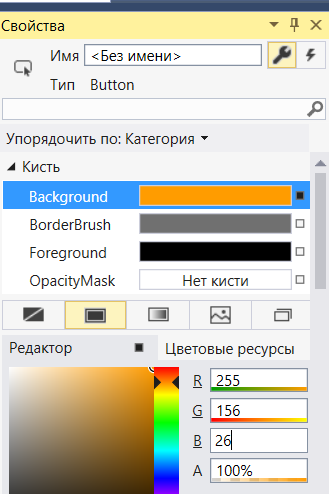
8. Откроется макет окна страницы ServicePage.

Поместим на форму кнопку:



Закрасим кнопку в активный цвет:

зададим свойство background и для него определим вручную цвет через вкладку Кисть, палитру RGB (255, 156, 26):



В коде разметки нужное значение само подставится.

9. Теперь добавим также страницу для редактирования/добавления сервиса (AddEditPage).

Поместим на эту форму (страницу) элемент TextBlock:

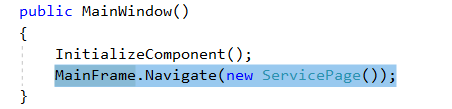


10. Вернемся к нашему фрейму на главной форме MainWindow.xaml, дадим ему имя:



11. Перейдем в редактор кода главной формы MainWindow.xaml.cs (нажать F7).

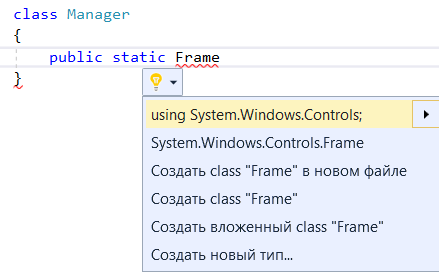
Добавим в описание класса после строки инициализации компонентов строку для навигации: MainFrame.Navigate(new ServicePage());



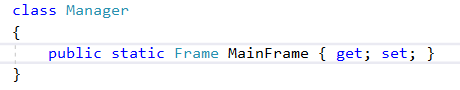
Но пока она не работает.

12. Поработаем с навигацией. Добавим класс Manager: правой кнопкой по названию решения (проекта) в Обозревателе решений, затем пункт Добавить 🡪 Класс.

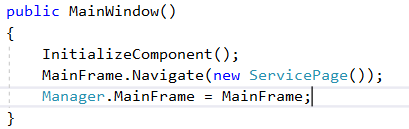
13. Опишем фрейм (когда будете писать при возникновении ошибки, нажмите на лампочку и выберите первый пункт для решения проблемы:



Далее дописать: MainFrame { get; set; }



14. Далее перейдем в окно редактора кода главной формы и допишем строчку



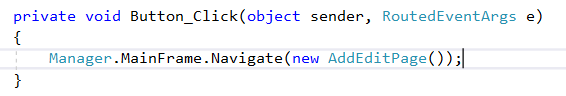
15. Перейдем на форму ServicePage.

Для кнопки добавим свойство Click – обработчик события – нажатие на кнопку:

Имя события само автоматически проставится: Click="Button\_Click"

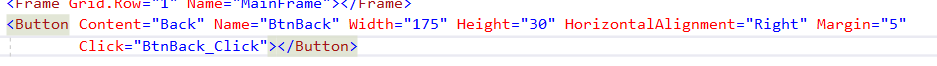
16. Перейдем в окно редактора кода этой страницы, в код для кнопки:

Запишем в коде одну строчку

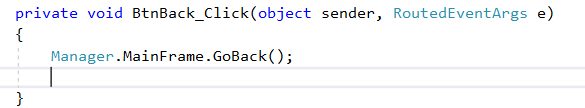


17. Запустим приложение. Проверить работу.

18. Добавим на главную форму кнопку Back:



19. Перейдем в код обработчика события для этой кнопки (F12) и запишем строчку

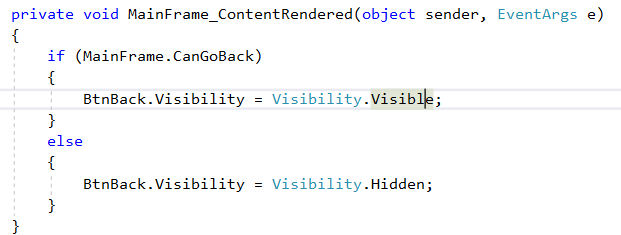


Для того, чтобы скрыть кнопку назад на главной форме используем свойство contentrendered у фрейма на главной форме:





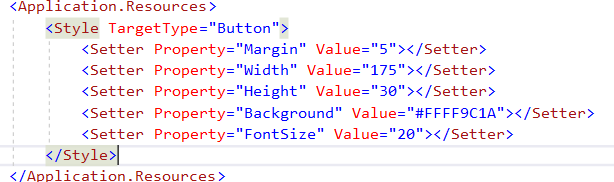
20. В обработчике пропишем:



21. Скроем панель навигации. Для этого у фрейма пропишем свойство:

NavigationUIVisibility="Hidden"

22. Создадим стили для элемента кнопка, чтобы при создании новых кнопок их внешний вид был одинаков. Для этого откроем App.xaml и пропишем:



И удалим везде по файлам разметки лишнее (ширину, высоту кнопок и тд)

23. Запустим, проверим.

Результат выполнения лаб. работы в архив, и в СДО.