[Początek](#a) (podstawowe informacje, przygotowanie środowiska Visual Studio 2017 + Xamarin)

[1.Wyświetlenie tekstu na nowej własnej stronie](#one)

[2.Dodanie przycisku i zadeklarowanie działania](#two)

[3.Wygląd UI](#three)

4.[Dodanie przycisku nawigacji do następnej strony](#four)

[5.Przekazanie informacji do następnej strony](#five)

[6.Wyświetlanie stron jako "Tabbed" i "Carousel"](#six)

[7.Widok - Lista](#seven)

[Aplikacje](#Aplikacje)

[Twórcy](#twórcy)

[Początek](#_top)

// Z początku pojawi się około 31 błędów - są ponieważ nic nie jest zadeklarowane w aplikacji portable

// NIE "NAPRAWIAJ BŁĘDÓW" - zniszczysz aplikacje i nie będzie ona xross platformowa

// Prawy przycisk myszy na Rozwiązanie "nazwa twojej aplikacji" -> zarządzaj pakietami nuget rozwiązania ->

// zainstaluj wszystkie aktuazlizacje z źródła M.V.S. Offline.. i aktualizację xamarin z nuget

// po pierwsze upewnij się o działalności emulatora androida (intel atom x88 - 10 x szybciej działa od arm)

// jeżeli nie - uruchom CMD jako administrator, wpisz "bcdedit /set hypervisorlaunchtype off" i uruchom komputer ponownie

//Jeżeli nie działa UWP na maszynie „lokalnej” - nacisnij prawym przyciskiem myszy na „nazwa twojej apki”.UWP -> wdróż projekt -> uruchom na maszynie lok (ctrl + F5)

// aby uruchomić projekt w kilku środowiskach - Prawy przycisk myszy na Rozwiązanie "nazwa twojej aplikacji" -> ustaw projekty startowe -> w zakładce wiele projektów startowych ... "uruchom"...

JEŻELI WSZYSTKO POPRAWNIE ZROBIŁEŚ PO URUCHOMIENIU APLIKACJI W TRYBIE DEBUG POJAWI SIĘ NA MEULATORZE NAPIS Welcome to Xamarin Forms

/// inne podstawowe informacje

Aplikacje Cross-Platformowe Visual Studio 2017 + Xamarin

XAML - Extensible Application Markup Language

Komentarz w języku C

// text w c

Komentarz w języku XAML

<!-- tekst w xaml ... -->

Deklaracja tekstu XAML

<Label Text="wpisz swoj text"

HorizontalOptions="Center"

VerticalOptions="Center"

/>

HorizontalOptions = ” ” - opcje poziome

VerticalOptions = ” ” - opcje pionowe

[1.Wyświetlenie tekstu na nowej własnej stronie](#_top)

„Pusta aplikacja” Nazwa aplikacji AppXross1

W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->

Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Domowa”

App.xaml.cs wpisz

public App ()

{

MainPage = new Domowa(); //Zadeklarowanie naszej strony jako domowej

//InitializeComponent(); pominelem

//MainPage = new AppXross1.MainPage(); pominelem

}

Domowa.xaml wpisz :

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

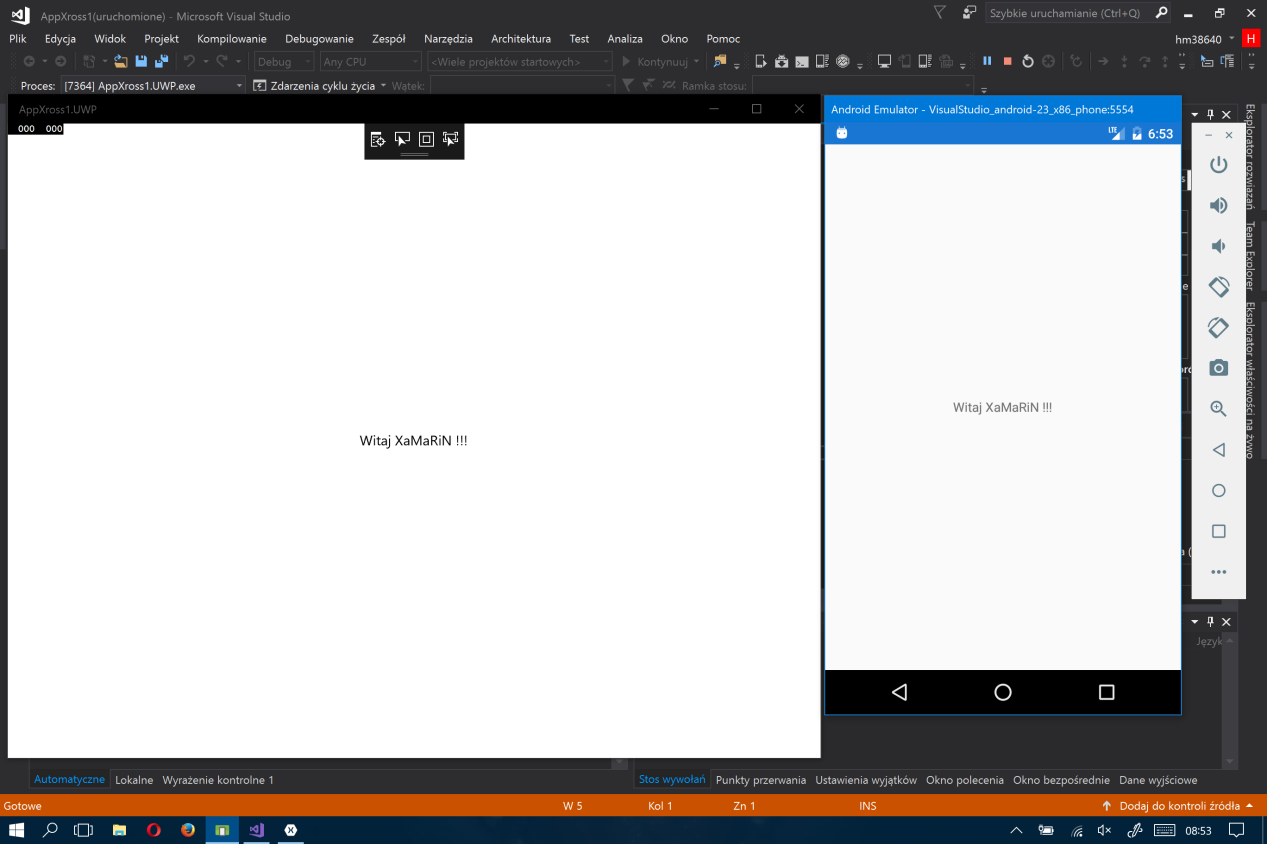
xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="AppXross1.Domowa">

<Label Text="Witaj XaMaRiN !!!" VerticalOptions="Center" HorizontalOptions="Center" /> <!--Zadeklarowany text w xaml-->

</ContentPage>

Symulacja wyjścia w trybie debug - UWP(maszyna lokalna),Android 6.0



[2.Dodanie przycisku i zadeklarowanie działania](#_top)

„Pusta aplikacja” Nazwa aplikacji AppXross1

W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->

Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Domowa”

App.xaml.cs wpisz

public App ()

{

MainPage = new Domowa(); //Zadeklarowanie naszej strony jako domowej

//InitializeComponent(); pominelem

//MainPage = new AppXross1.MainPage(); pominelem

}

Domowa.xaml

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="AppXross1.Domowa">

<StackLayout>

<!-- powłoka w ktorej bdzie realizowane to co jest wewnatrz -->

<Label Text="Podaj swoje imię"/>

<!-- Deklaruje widoczny tekst na samej górze aplikacji-->

<Entry x:Name="GlownyWpis"/>

<!-- Nasze pole tekstowe w ktorym napiszemy swoje imie -->

<Button Clicked="Button\_Clicked"

Text="Przywitaj się!"/>

<!-- przycisk w xaml-->

<!-- Po wpisaniu Clicked=" <nowa procedura obsługi zdarzeń>"

zostanie utworzona w Domowa.xaml.cs funkcja określająca działanie przycisku -->

<Label x:Name="GlownyLabel"/>

<!-- będzie realizować dane zadanie -> patrz Domowa.xaml.cs

(mozesz tez kliknac prawym przyciskiem myszy -> przejdz do definicji) -->

</StackLayout>

</ContentPage>

Domowa.xaml.cs

private void Button\_Clicked(object sender, EventArgs e)

{

string text = GlownyWpis.Text;

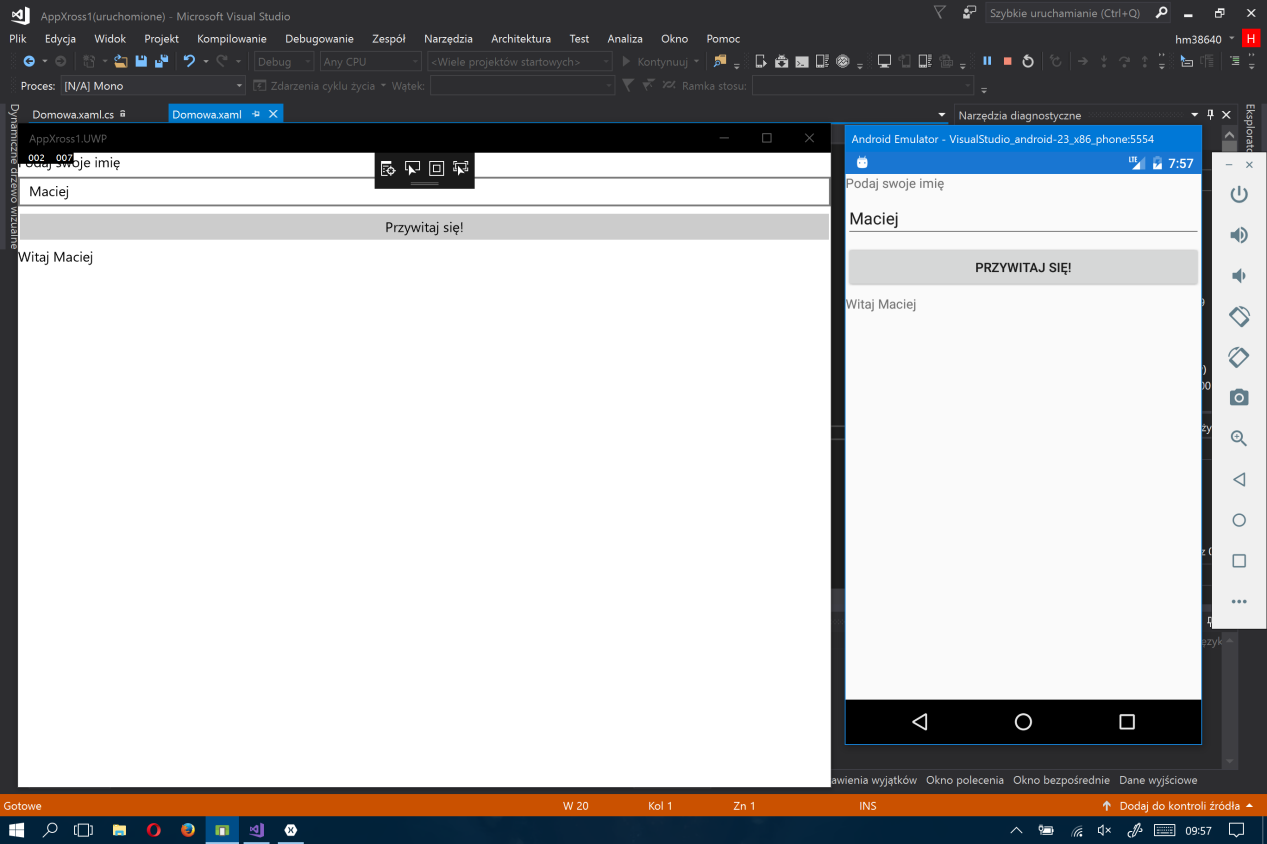
//odniesienie do entry x.Name

GlownyLabel.Text = "Witaj " + text;

//odniesienie do Label x:Name="GlownyLabel" + powyzszy string text

}

Symulacja wyjścia w trybie debug - UWP(maszyna lokalna),Android 6.0



[3.Wygląd UI](#_top)

[„Pusta aplikacja” Nazwa aplikacji AppXross1](#_top)

[W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->](#_top)

[Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Domowa”](#_top)

[App.xaml.cs wpisz](#_top)

[public App ()](#_top)

[{](#_top)

[MainPage = new Domowa(); //Zadeklarowanie naszej strony jako domowej](#_top)

[//InitializeComponent(); pominelem](#_top)

[//MainPage = new AppXross1.MainPage(); pominelem](#_top)

[}](#_top)

[Domowa.xaml](#_top)

[<?xml version="1.0" encoding="utf-8" ?>](#_top)

[<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"](#_top)

[xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"](#_top)

[x:Class="AppXross1.Domowa"](#_top)

[BackgroundColor="White">](#_top)

[<!-- Bez BackgroundColor o takim samym kolorze co stack layaut](#_top)

[pojawią się paski o innych kolorach (chyba że nie zastosuje się VerticalOptions w stacklayaut) -->](#_top)

[<StackLayout BackgroundColor="White"](#_top)

[Padding="60"](#_top)

[VerticalOptions="Center">](#_top)

[<!--Kolor powłoki aplikacji-->](#_top)

[<!--Padding powoduje, że przyciski i tekst oddalą się od krawędzi ekranu w ""-->](#_top)

[<!--VerticalOptions - centruje powłokę aplikacji (Bez BackgroundColor będą paki) -->](#_top)

[<!-- powłoka w ktorej bdzie realizowane to co jest wewnatrz -->](#_top)

[<Label Text="Podaj swoje imię"/>](#_top)

[<!-- Deklaruje widoczny tekst na samej górze aplikacji-->](#_top)

[<Grid Padding="0,0,0,50">](#_top)

[<!-- Padding nadaje przerwę od głównego wpisu "0,0,0,50" lewo, góra, prawo, dół-->](#_top)

[<Entry x:Name="GlownyWpis"](#_top)

[BackgroundColor="Gray"/>](#_top)

[<!--Kolor wpisu-->](#_top)

[</Grid>](#_top)

[<!-- Nasze pole tekstowe w ktorym napiszemy swoje imie -->](#_top)

[<Button Clicked="Button\_Clicked"](#_top)

[Text="Przywitaj się!"](#_top)

[BackgroundColor="Green"](#_top)

[TextColor="White"/>](#_top)

[<!--Kolor przycisku i kolor tekstu-->](#_top)

[<!-- przycisk w xaml-->](#_top)

[<!-- Po wpisaniu Clicked=" <nowa procedura obsługi zdarzeń>"](#_top)

[zostanie utworzona w Domowa.xaml.cs funkcja określająca działanie przycisku -->](#_top)

[<Label x:Name="GlownyLabel"](#_top)

[TextColor="Gray"](#_top)

[FontSize="40"/>](#_top)

[<!--Kolor tekstu i rozmiar czcionki w "" -->](#_top)

[<!-- będzie realizować dane zadanie -> patrz Domowa.xaml.cs](#_top)

[(mozesz tez kliknac prawym przyciskiem myszy -> przejdz do definicji) -->](#_top)

[</StackLayout>](#_top)

[</ContentPage>](#_top)

[Domowa.xaml.cs](#_top)

[private void Button\_Clicked(object sender, EventArgs e)](#_top)

[{](#_top)

[string text = GlownyWpis.Text;](#_top)

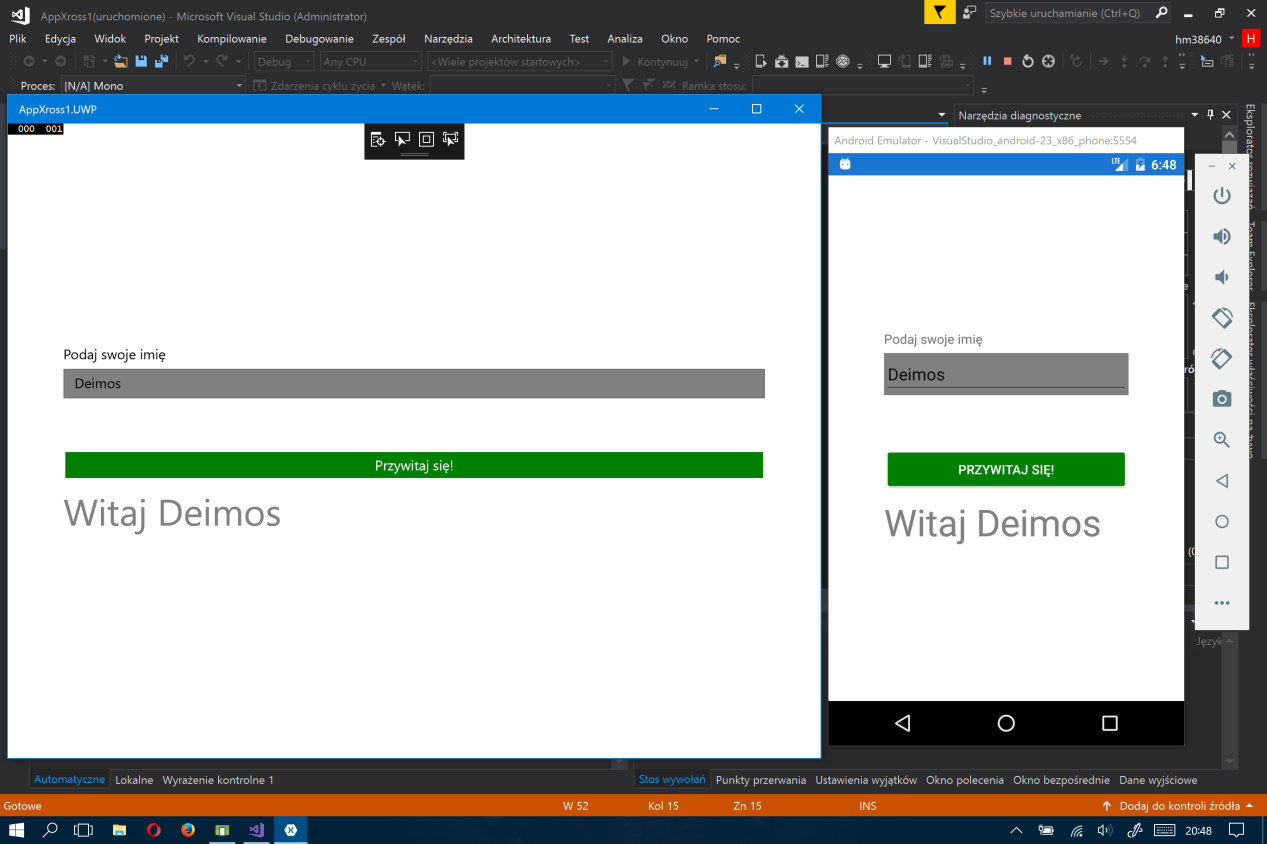
[//odniesienie do entry x.Name](#_top)

[GlownyLabel.Text = "Witaj " + text;](#_top)

[//odniesienie do Label x:Name="GlownyLabel" + powyzszy string text](#_top)

[}](#_top)

Symulacja wyjścia w trybie debug - UWP(maszyna lokalna),Android 6.0



1. [Dodanie przycisku nawigacji do następnej strony](#_top)

[„Pusta aplikacja” Nazwa aplikacji AppXross1](#_top)

[W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->](#_top)

[Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Domowa”](#_top)

[W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->](#_top)

[Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Page1”](#_top)

[Page1.xaml](#_top)

[<?xml version="1.0" encoding="utf-8" ?>](#_top)

[<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"](#_top)

[xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"](#_top)

[x:Class="AppXross1.Page1"](#_top)

[Title="Druga strona">](#_top)

[<!--Nadaliśmy tytuł naszej stronki -->](#_top)

[</ContentPage>](#_top)

[Domowa.xaml](#_top)

[<?xml version="1.0" encoding="utf-8" ?>](#_top)

[<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"](#_top)

[xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"](#_top)

[x:Class="AppXross1.Domowa"](#_top)

[BackgroundColor="White">](#_top)

[<StackLayout BackgroundColor="White"](#_top)

[Padding="60"](#_top)

[VerticalOptions="Center">](#_top)

[<Label Text="Podaj swoje imię"/>](#_top)

[<Grid Padding="0,0,0,50">](#_top)

[<Entry x:Name="GlownyWpis"](#_top)

[BackgroundColor="Gray"/>](#_top)

[</Grid>](#_top)

[<Button Clicked="Button\_Clicked"](#_top)

[Text="Przywitaj się!"](#_top)

[BackgroundColor="Green"](#_top)

[TextColor="White"/>](#_top)

[<Label x:Name="GlownyLabel"](#_top)

[TextColor="Gray"](#_top)

[FontSize="40"/>](#_top)

[<Button Text="Przejdź do drugiej strony"](#_top)

[Clicked="Button\_Clicked\_1"](#_top)

[BackgroundColor="Lavender">](#_top)

[<!--tekst na przycisku + nowa definicja zdarzenia Clicked tj Button\_Clicked\_1 -->](#_top)

[</Button>](#_top)

[</StackLayout>](#_top)

[</ContentPage>](#_top)

[Domowa.xaml.cs](#_top)

[private void Button\_Clicked(object sender, EventArgs e)](#_top)

[{](#_top)

[string text = GlownyWpis.Text;](#_top)

[//odniesienie do entry x.Name](#_top)

[GlownyLabel.Text = "Witaj " + text;](#_top)

[//odniesienie do Label x:Name="GlownyLabel" + powyzszy string text](#_top)

[}](#_top)

[private async void Button\_Clicked\_1(object sender, EventArgs e) //musi być asynchroniczne z powodu metody await](#_top)

[{](#_top)

[await Navigation.PushAsync(new Page1()); //działanie naciśnięcia przycisku -> przekierowuje do Page1](#_top)

[}](#_top)

[App.xaml.cs](#_top)

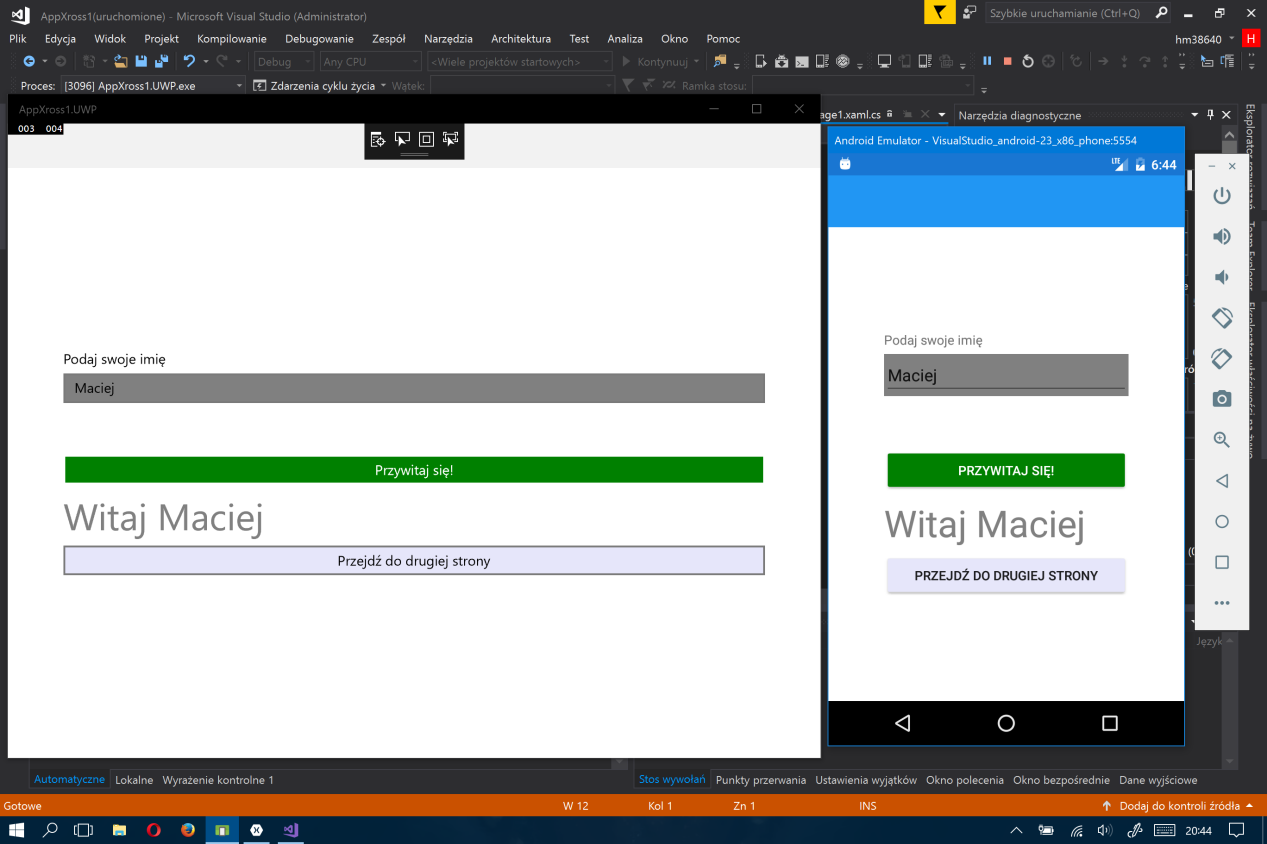
[public App ()](#_top)

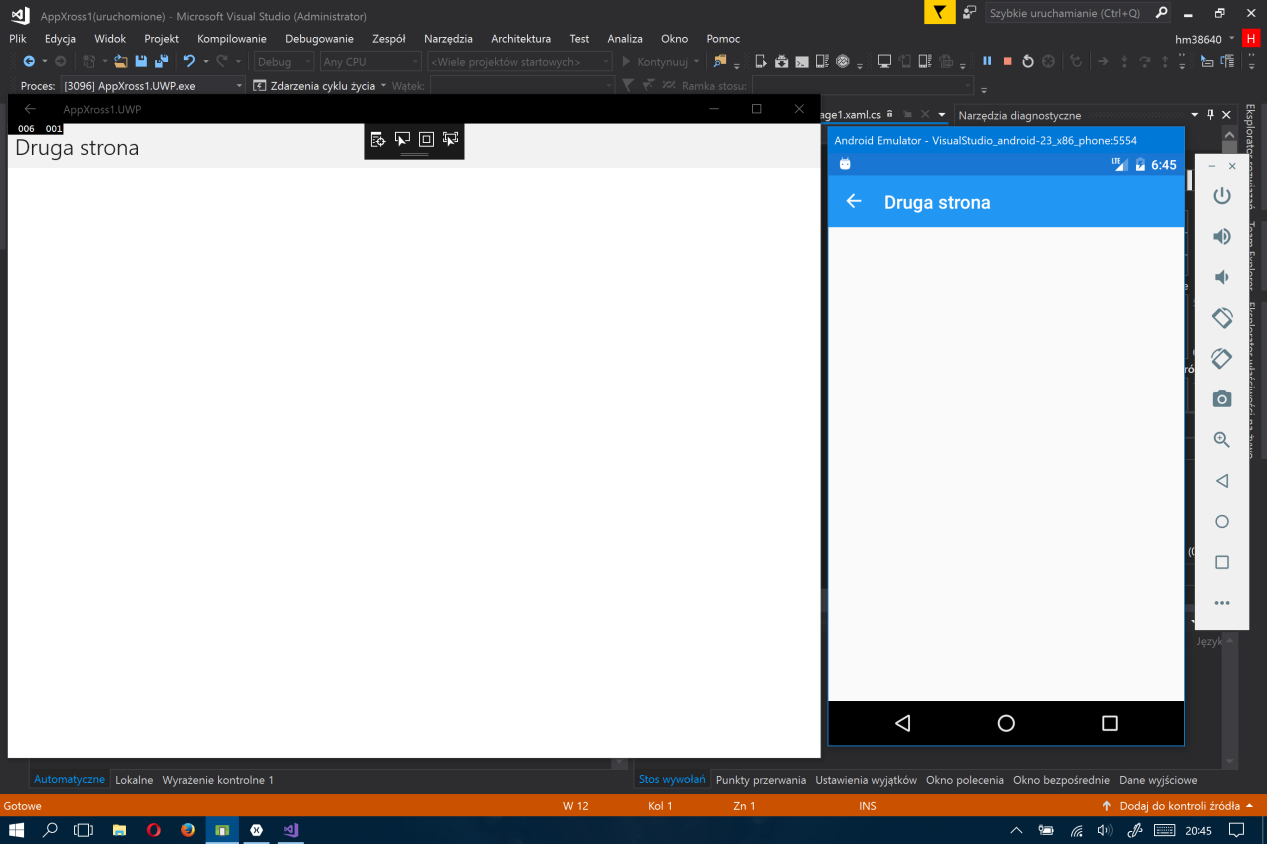
[{](#_top)

[MainPage = new NavigationPage(new Domowa()); //Zadeklarowanie naszej strony jako domowej](#_top)

[}](#_top)

[Symulacja wyjścia w trybie debug - UWP(maszyna lokalna),Android 6.0](#_top)

[[](#_top)](#_top)

[druga strona[](#_top)](#_top)

[5.Przekazanie informacji do następnej strony](#_top)

„Pusta aplikacja” Nazwa aplikacji AppXross1

W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->

Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Domowa”

W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->

Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Page1”

Domowa.xaml

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="AppXross1.Domowa"

BackgroundColor="White">

<StackLayout BackgroundColor="White"

Padding="60"

VerticalOptions="Center">

<Label Text="Podaj swoje imię"/>

<Grid Padding="0,0,0,50">

<Entry x:Name="GlownyWpis"

BackgroundColor="Gray"/>

<!--Parametr z GlownyWpis (to co wpiszemy w szarym obszarze)

zostanie przekazany na następną stronę -> patrz Domowa.xaml.cs -->

</Grid>

<Button Clicked="Button\_Clicked"

Text="Przywitaj się!"

BackgroundColor="Green"

TextColor="White"/>

<Label x:Name="GlownyLabel"

TextColor="Gray"

FontSize="40"/>

<Button Text="Przejdź do drugiej strony"

Clicked="Button\_Clicked\_1"

BackgroundColor="Lavender">

<!--tekst na przycisku + nowa definicja zdarzenia Clicked tj Button\_Clicked\_1 -->

</Button>

</StackLayout>

</ContentPage>

Domowa.xaml.cs

private void Button\_Clicked(object sender, EventArgs e)

{

string text = GlownyWpis.Text;

//odniesienie do entry x.Name

GlownyLabel.Text = "Witaj " + text;

//odniesienie do Label x:Name="GlownyLabel" + powyzszy string text

}

private async void Button\_Clicked\_1(object sender, EventArgs e) //musi być asynchroniczne z powodu metody await

{

await Navigation.PushAsync(new Page1(GlownyWpis.Text));

//działanie naciśnięcia przycisku -> przekierowuje do Page1

//wpisany tekst z GlownyWpis (patrz -> domowa.xaml) zostanie przekazany do Page1

}

App.xaml.cs

public App ()

{

MainPage = new NavigationPage(new Domowa());

//Zadeklarowanie naszej strony jako domowej

//Strona Nawigacyjna jest zadeklarowana jako podstrona domowej

}

Page1.xaml

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="AppXross1.Page1"

Title="Druga strona">

<!--Nadaliśmy tytuł naszej stronki -->

<Label x:Name="DrogiLabel"

TextColor="Gray"/>

<!--Label wyświetla text w Xamarin.Forms-->

<!--Zadeklarowujemy go jako parametr w Page1.xaml.cs-->

</ContentPage>

Page1.xaml.cs

public Page1 (string parameter)

//A string is a sequential collection of Unicode characters that is used to represent text.

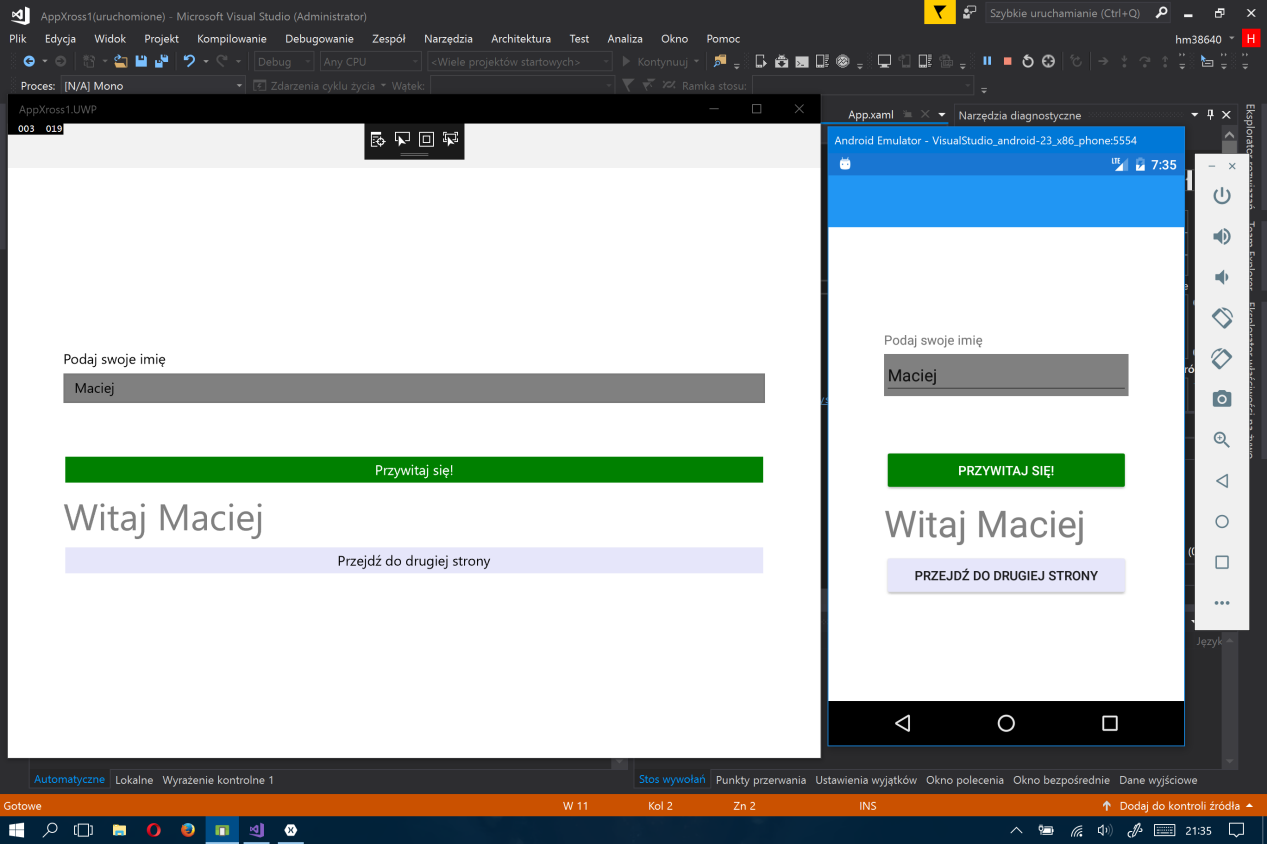
//Więcej informacji na temat stringa jako klasy i konstruktora na https://developer.xamarin.com/api/type/System.String/

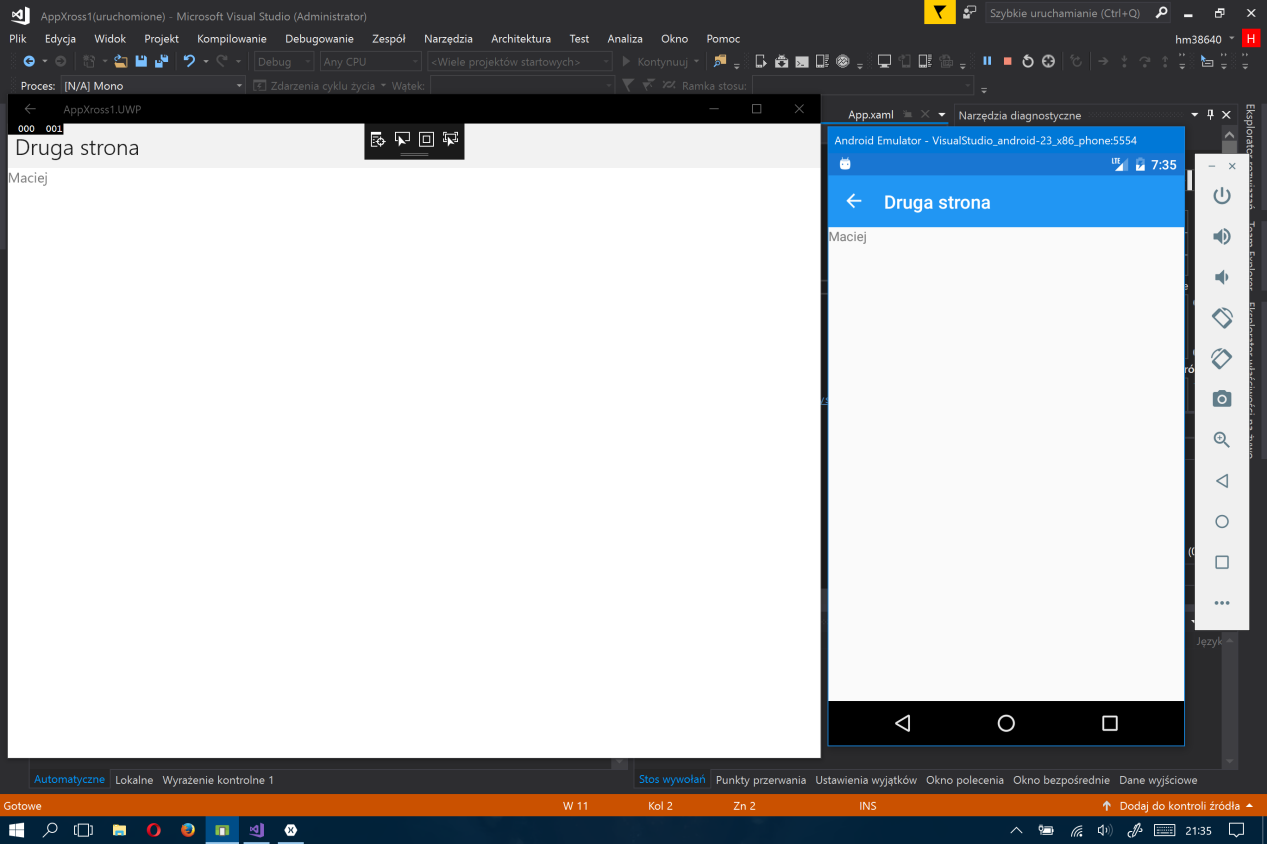
{

InitializeComponent ();

DrogiLabel.Text = parameter; //odnośnik od Label w Page1.xaml jako parametru

}





[6.Wyświetlanie stron jako "Tabbed" i "Carousel"](#_top)

->Aby wyświetlać strony w trybie „Tabbed” należy w aplikacji z pkt 5 zmienić

App.xaml.cs

public App ()

{

MainPage = new TabbedPage //zadeklarowanie strony "tabowanej"

{

Children = //zadeklarowanie stron jako zakładek

{

new Domowa(),

new Page1("Witaj !!!")

}

};

}

Domowa.xaml

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="AppXross1.Domowa"

BackgroundColor="White"

Title="Strona Domowa"> <!--Zostało dodane -->

->Aby wyświetlać strony w trybie „Carousel” należy w aplikacji z pkt 5 zmienić

App.xaml.cs

public App ()

{

MainPage = new CarouselPage//zadeklarowanie strony "Carousel"

{

Children = //zadeklarowanie stron jako zakładek

{

new Domowa(),

new Page1("Witaj !!!")

}

};

}

Domowa.xaml

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

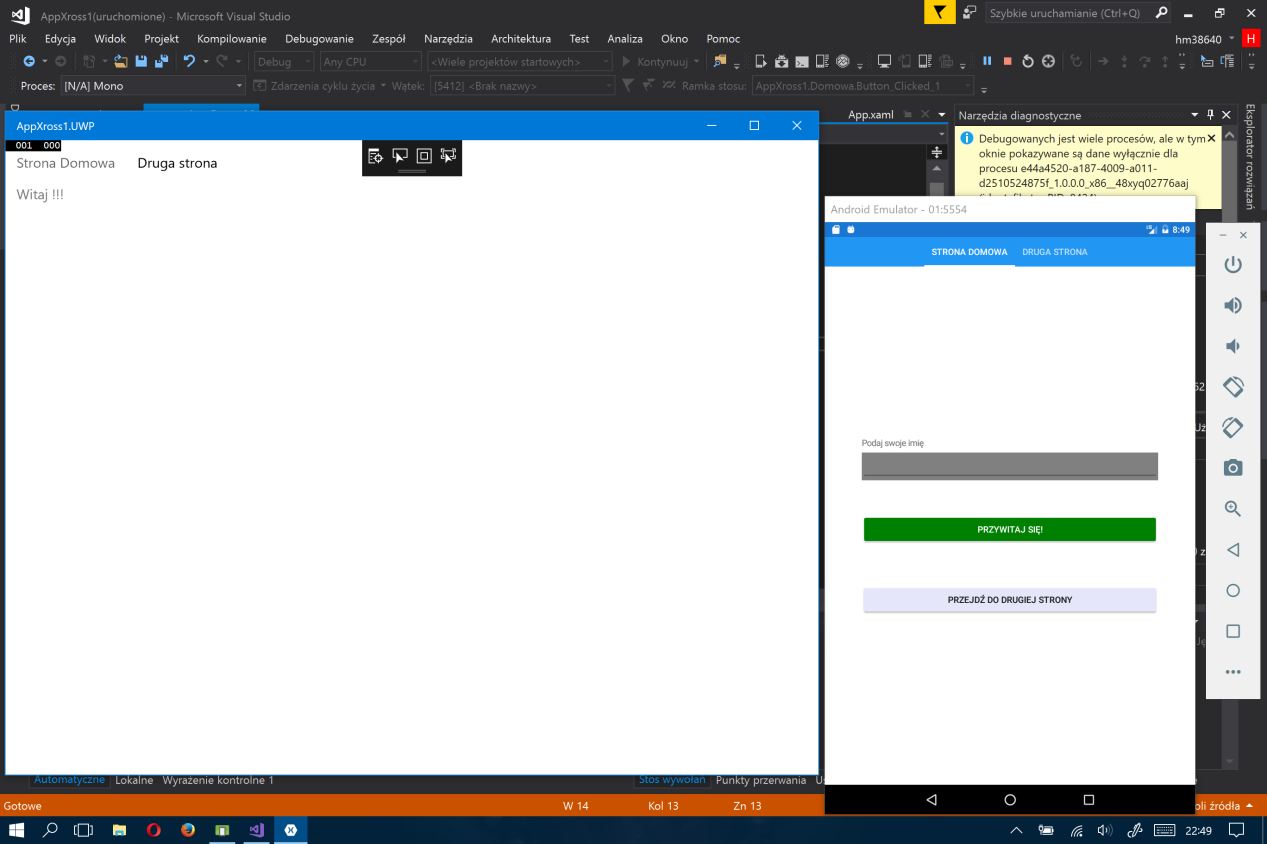
xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="AppXross1.Domowa"

BackgroundColor="White"

Title="Strona Domowa"> <!--Zostało dodane -->

Strona tabowana



[7.Widok - Lista](#_top)

„Pusta aplikacja” Nazwa aplikacji AppXross7

W Eksploratorze Rozwiązań naciśnij prawym przyciskiem myszy na AppXross1 ->

Dodaj -> nowy element -> Cross-Platform -> Forms Blank Content Page Xaml i nazwij go „Domowa”

Domowa.xaml

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="AppXross7.Domowa"

BackgroundColor="White"

Title="Strona Domowa">

<!--Kolor powłoki "BackgroundColor" - biały

Nazwa storny "Title"-->

<ListView x:Name="GlownyListView" />

<!--Dodajemy Widok listowy pod daną nazwą

i deklarujemy w Domowa.xaml.cs-->

</ContentPage>

Domowa.xaml.cs

public Domowa ()

{

InitializeComponent ();

GlownyListView.ItemsSource = new List<string>

{

"Maciej", "Deimos", "Mojżesz", "3perełki ;)", "Jestem", "Byłem", "Będę"

};

//przypomnienie info o string : https://developer.xamarin.com/api/type/System.String/

//info o itemssource https://developer.xamarin.com/api/property/Xamarin.Forms.ItemsView<TVisual>.ItemsSource/

}

App.xaml.cs

public App ()

{

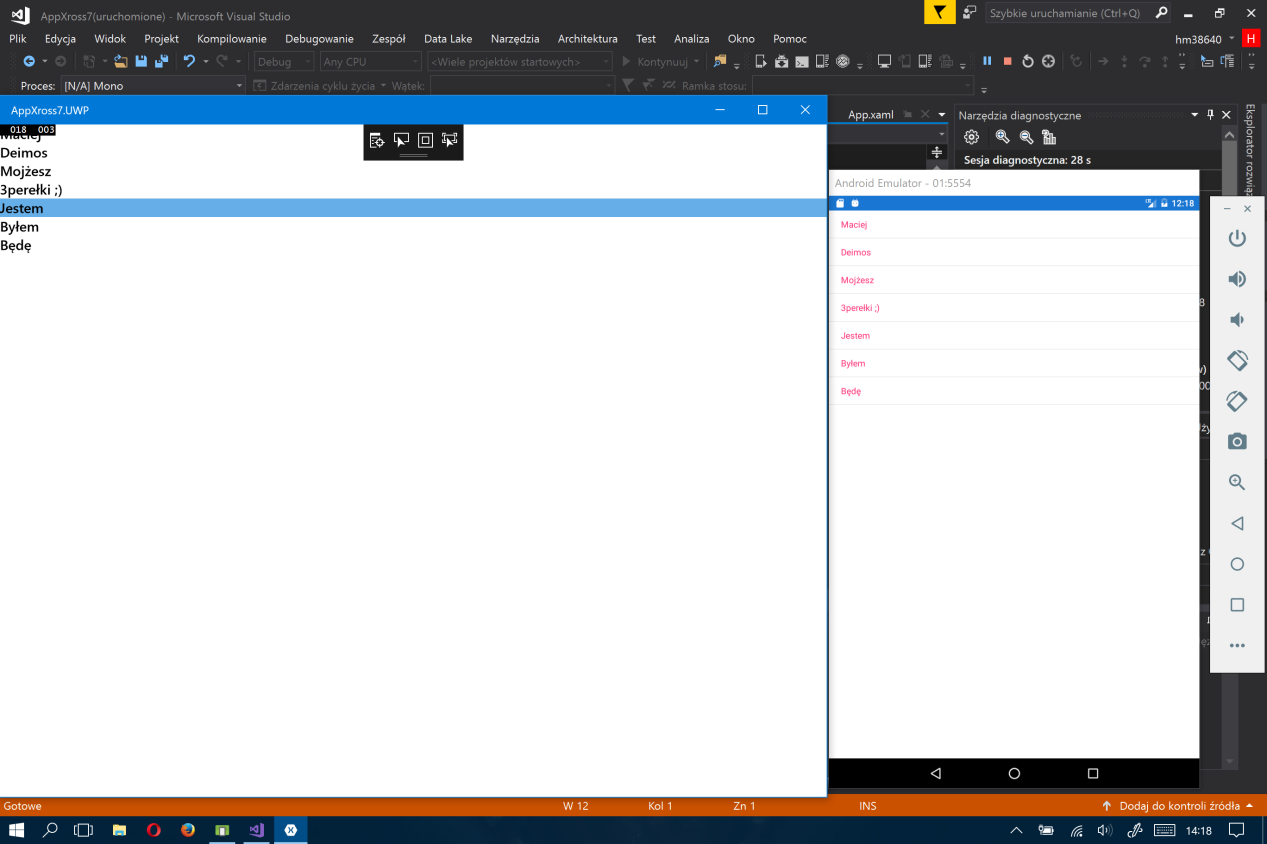
MainPage = new Domowa();

//InitializeComponent(); pomiń

//MainPage = new AppXross7.MainPage(); pomiń

}

Wynik



[Aplikacje](#_top)

[1](Aplikacje/AppXross1)

[2](Aplikacje/AppXross2)

[3](Aplikacje/AppXross3)

[4](Aplikacje/AppXross4)

[5](Aplikacje/AppXross5)

[6](Aplikacje/AppXross6)

[7](Aplikacje/AppXross7)

*[Created by](#_top):*

*Maciej Hejlasz*

Facebook

