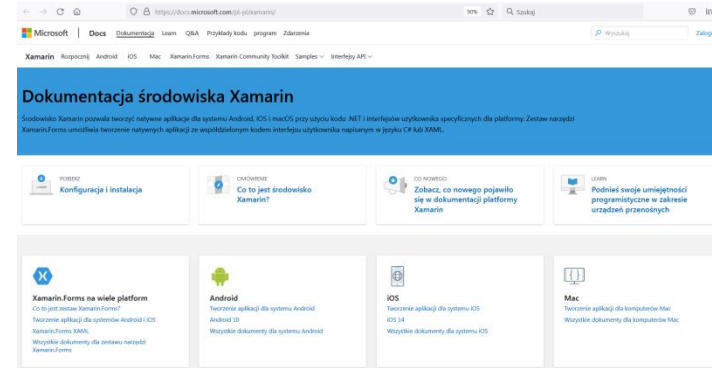


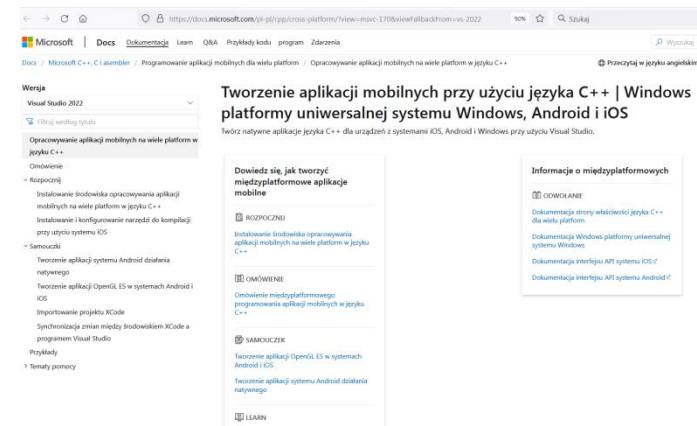
aplikacje mobilne

Aplikacje mobilne - Visual Studio

Tworzenie aplikacji mobilnych przy użyciu platformy .NET (Xamarin)



Tworzenie aplikacji mobilnych w języku C++



Learn about mobile development with JavaScript in Visual Studio 2017

Artykuł • 31.10.2016 • Czas czytania: 2 min •

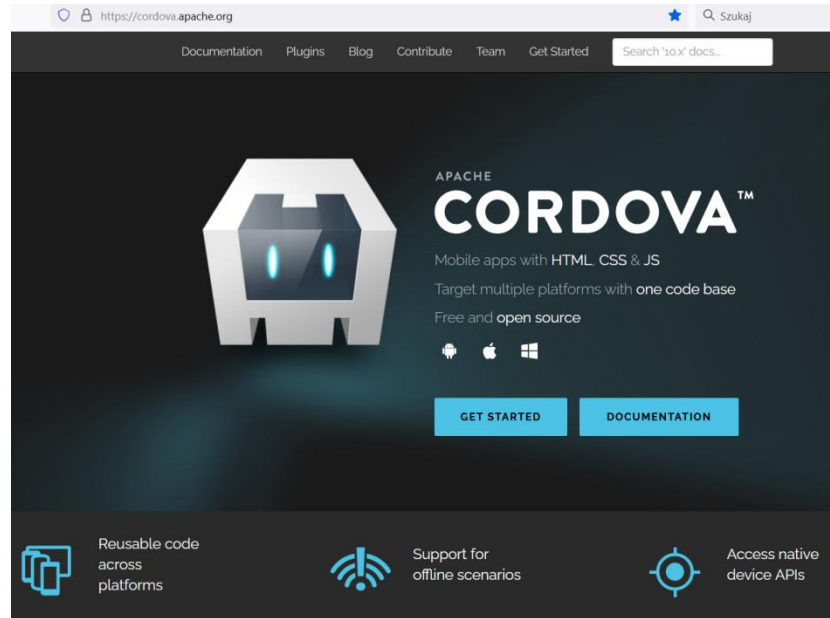
Overview

Microsoft's Visual Studio Tools for Apache Cordova is a Microsoft Visual Studio workload that simplifies development of mobile apps on Android, iOS, and Windows using [Apache Cordova](#). Apache Cordova enables cross-platform mobile development using standard web technologies (HTML, CSS, and JavaScript). The Tools for Apache Cordova may be installed along with your initial Visual Studio installation, or added to it later by simply re-running the installer. The workload installs a complete Apache Cordova development environment within Visual Studio, plus adds Cordova project templates and Cordova-specific capabilities to the IDE. It includes the following capabilities:

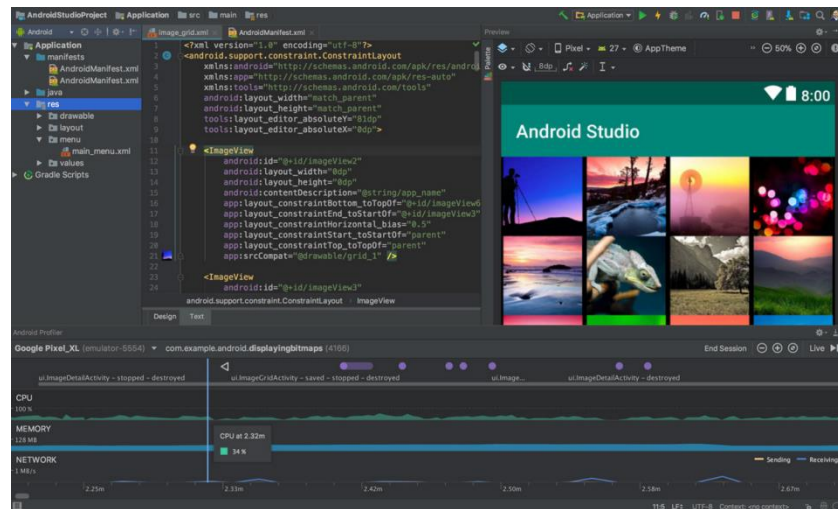
Tworzenie aplikacji mobilnych w języku JavaScript (w oparciu o Apache Cordova tylko Visual Studio 2017 and 2015)

Aplikacje mobilne – różne technologie

Tworzenie aplikacji mobilnych
w oparciu o HTML, CSS oraz JS

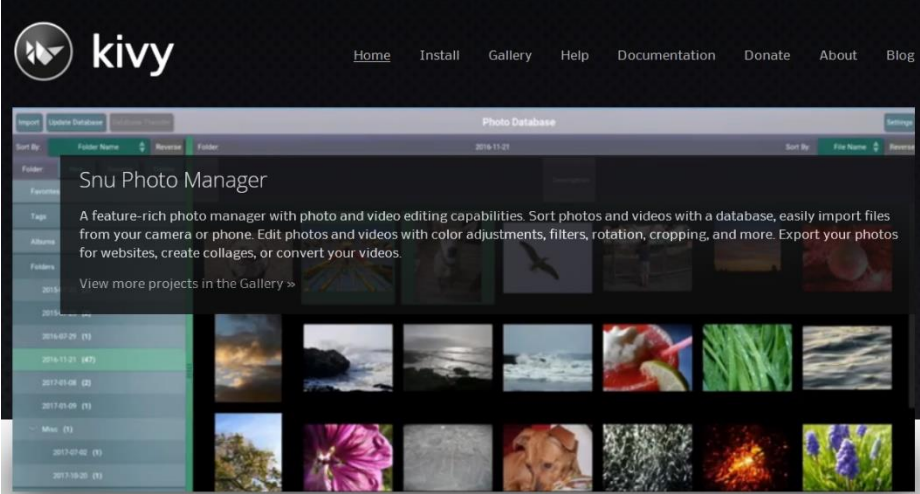


Android Studio



Aplikacje mobilne – różne technologie

biblioteka języka Python



The screenshot displays the Kivy website with a dark theme. At the top, the Kivy logo is on the left, and navigation links (Home, Install, Gallery, Help, Documentation, Donate, About, Blog) are on the right. Below the navigation is a 'Photo Database' section with a sidebar menu (Folders, Favorites, Tags, Albums, Folders) and a main content area. The main content area features the title 'Snu Photo Manager' and a description: 'A feature-rich photo manager with photo and video editing capabilities. Sort photos and videos with a database, easily import files from your camera or phone. Edit photos and videos with color adjustments, filters, rotation, cropping, and more. Export your photos for websites, create collages, or convert your videos.' Below the text is a grid of photo thumbnails. At the bottom of the screenshot, there is a text block: 'Kivy - Open source Python library for rapid development of applications that make use of innovative user interfaces, such as multi-touch apps.' and three icons with text: 'Cross platform', 'Business Friendly', and 'GPU Accelerated'.

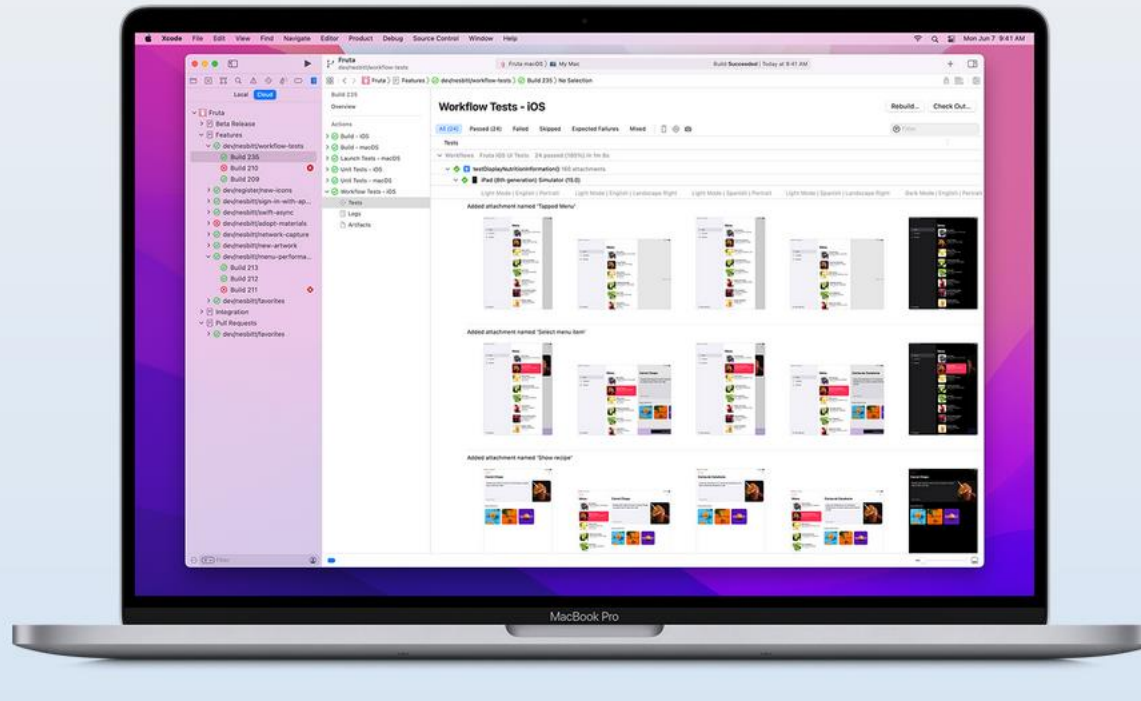
Kivy - Open source Python library for rapid development of applications that make use of innovative user interfaces, such as multi-touch apps.

- Cross platform
- Business Friendly
- GPU Accelerated

Aplikacje mobilne – różne technologie

Xcode 13

Xcode 13 includes the latest SDKs for iOS 15, iPadOS 15, tvOS 15, watchOS 8, and macOS Monterey.



Instalacja Visual Studio Community - Xamarin

<https://visualstudio.microsoft.com/pl/>



Microsoft | Visual Studio | Produkty | Materiały do pobrania | Kup | Pomoc techniczna | Dostęp dla subskrybentów | **Bezpłatne Visual Studio** | Wszystkie produkty Microsoft | Wyszukaj | Zaloguj się

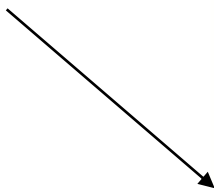
To jest sposób tworzenia oprogramowania

Nad czym chcesz dzisiaj [code, build, debug, deploy, współpracować, i co przeanalizować, learn]?

Visual Studio może to zrobić.

Poznaj rodzinę programu Visual Studio

instalator



Visual Studio
Wersja 17.
Najlepsze kompleksowe środowisko IDE dla deweloperów platformy .NET i języka C++ w systemie Windows. Pełna gama narzędzi i funkcji pozwalających podnieść poziom i ulepszyć każdy etap opracowywania oprogramowania.

Bezpłatnie dla indywidualnych deweloperów, zastosowań akademickich i rozwiązań typu open source

Pobierz Visual Studio

Community 2022

Professional 2022

Visual Studio for Mac
Wersja 8.10
Kompleksowe środowisko IDE dla deweloperów platformy .NET, które jest natywne dla systemu macOS. Obejmuje najwyższej klasy wsparcie dla sieci, chmury i tworzenia gier oraz bardzo dobre narzędzia do tworzenia międzyplatformowych aplikacji mobilnych.

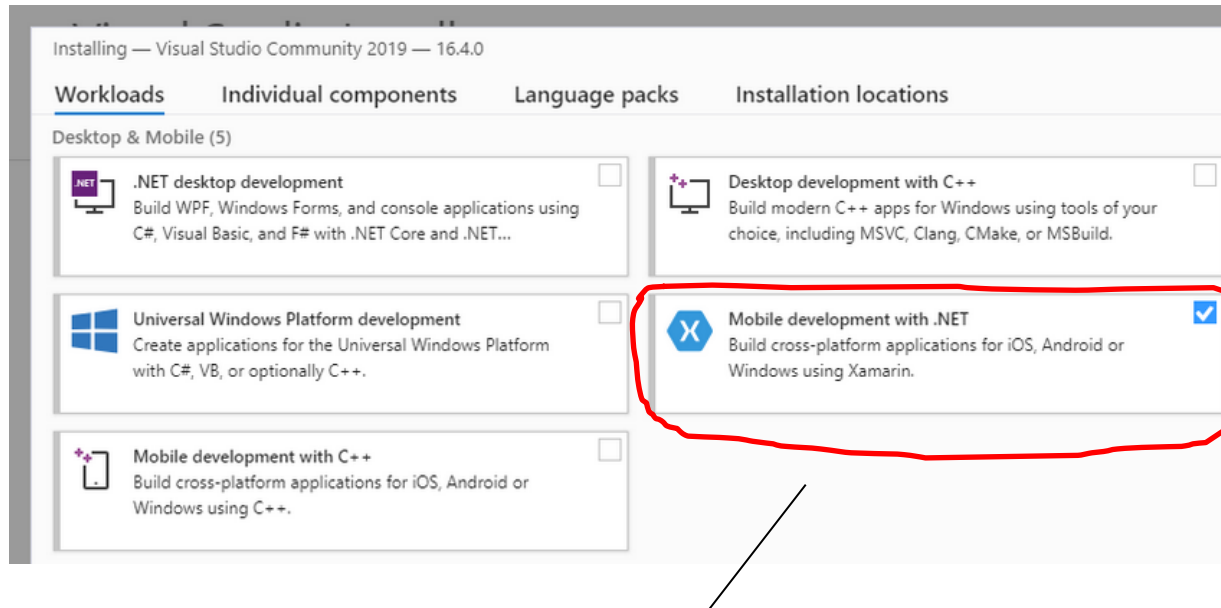
komputerów Mac

Visual Studio Code
Wersja 1.62
Autonomiczny edytor kodu źródłowego działający w systemach Windows, macOS i Linux. Najlepszy wybór dla deweloperów języka Java i internetu z mnóstwem rozszerzeń do obsługi niemal dowolnego języka programowania.

Korzystając z programu Visual Studio Code, akceptujesz jego licencja & oświadczenie o ochronie prywatności

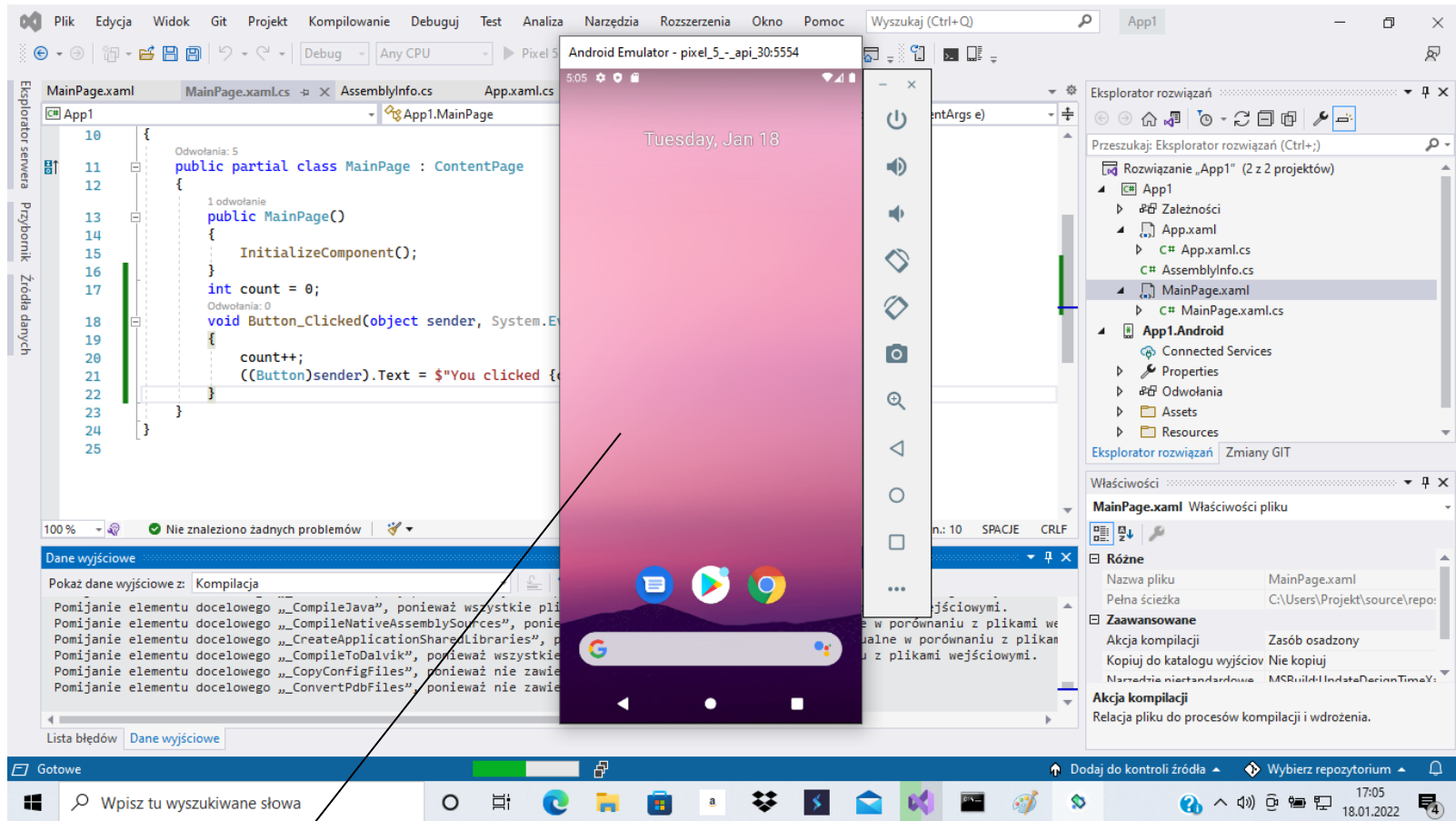
Pobierz program Visual Studio Code

Instalacja Visual Studio Community - Xamarin



Podczas instalacji należy wybrać moduł Xamarin

Xamarin



Po uruchomieniu programu (Ctrl F5) pojawia się emulator komórki z Androidem

Xamarin – hardware acceleration



Przyspieszanie sprzętowe wydajności emulatora & (HAXM funkcji Hyper-V)

Artykuł • 02.12.2021 • Czas czytania: 7 min

[Czy ta strona jest pomocna?](#)

Wybierz swoje Środowisko deweloperskie

Visual Studio 2019

Visual Studio for Mac

W tym artykule wyjaśniono, jak używać funkcji przyspieszania sprzętowego komputera w celu zmaksymalizowania wydajności Emulator Android.

Visual Studio ułatwia deweloperom testowanie i debugowanie aplikacji platformy Xamarin.Android przy użyciu emulatora systemu Android w sytuacjach, gdy urządzenie z systemem Android jest niedostępne lub niepraktyczne. Emulator systemu Android działa jednak zbyt wolno, jeśli przyspieszanie sprzętowe nie jest dostępne na komputerze, na który jest uruchamiany. Wydajność emulatora systemu Android można znacząco poprawić, korzystając ze specjalnych obrazów urządzeń wirtualnych x86 w połączeniu z funkcjami wirtualizacji komputera.

Scenariusz	HAXM	WHPX	Hypervisor.Framework
Masz procesor Intel	X	X	X
Masz procesor AMD		X	
Chcesz obsługiwać funkcji Hyper-V		X	
Chcesz obsługiwać zagnieżdżone wirtualizacji		Ograniczone	
Chcesz używać technologii takich jak Docker	(z WSL2)	X	X

Aby emulator komórki działał wydajniej można włączyć przyspieszenie sprzętowe (jeśli mamy taką możliwość)

Xamarin – hardware acceleration

```
CA Wiersz polecenia
Microsoft Windows [Version 10.0.19042.630]
(c) 2020 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\Projekt>sc query intelhaxm

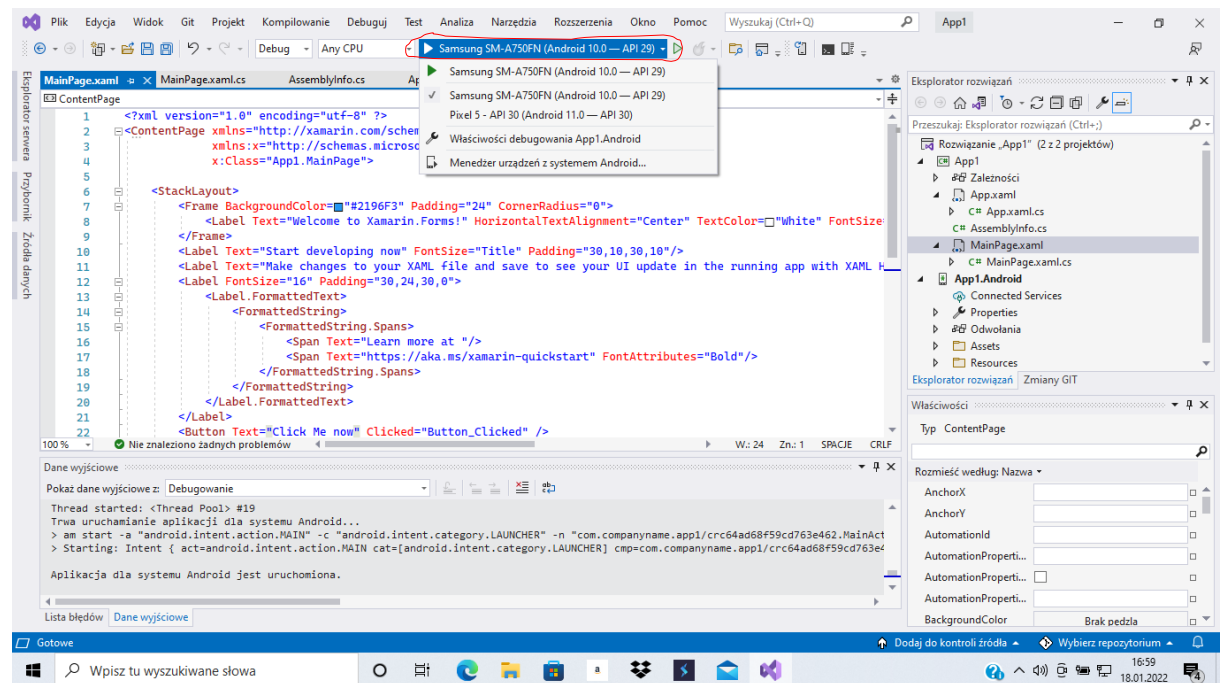
SERVICE_NAME: intelhaxm
        TYPE               : 1  KERNEL_DRIVER
        STATE                : 4  RUNNING
                        (STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        WIN32_EXIT_CODE       : 0  (0x0)
        SERVICE_EXIT_CODE    : 0  (0x0)
        CHECKPOINT           : 0x0
        WAIT_HINT            : 0x0

C:\Users\Projekt>_
```

W przypadku procesora Intel wyłączamy zabezpieczenie oparte na wirtualizacji oraz włączmy wirtualizację w BIOSie (Virtualization is Enabled). Następnie instalujemy program [wirtualizacji HAXM](#) i sprawdzamy czy działa poleceniem: `sc query intelhaxm`

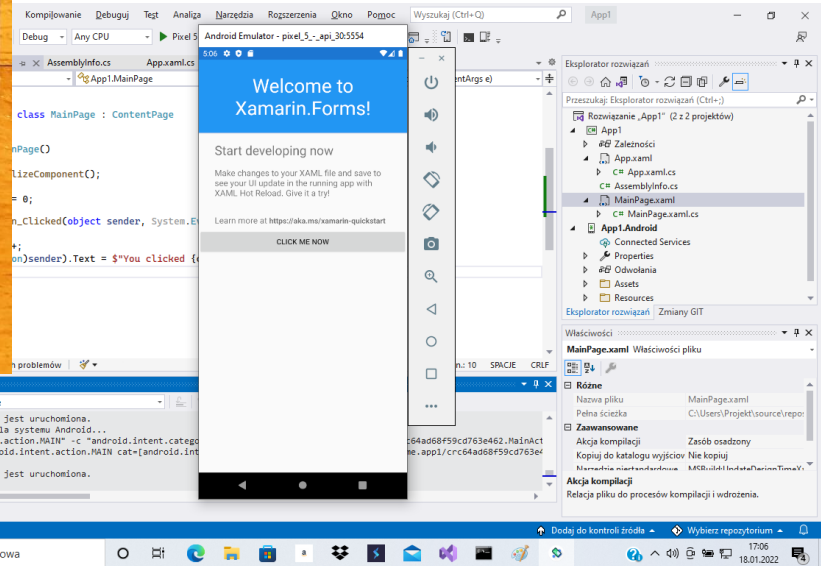
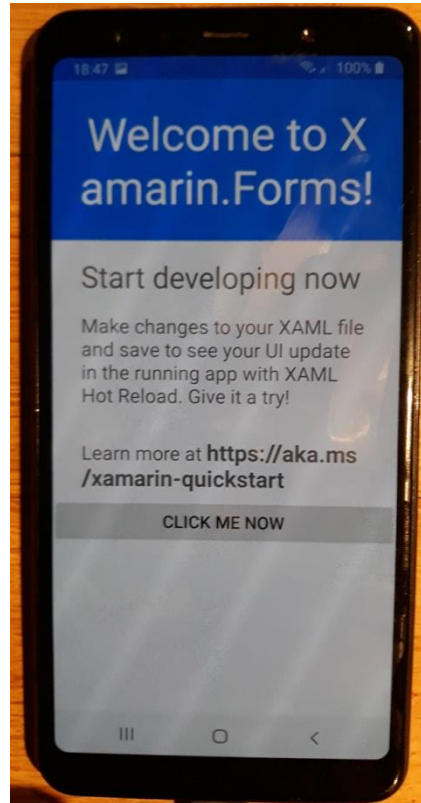
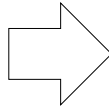
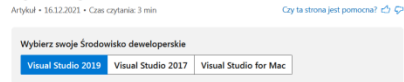
Xamarin – połączenie telefonu z Androidem

Komórkę z Androidem podłączmy za pomocą kabla USB do komputera, włączmy opcje debugowania na telefonie, wybieramy nasz aparat w menu. Od tej pory nasze aplikacje będą się uruchamiały na podłączonym telefonie. (dużo szybciej niż na emulatorze)



Xamarin – pierwsza aplikacja

Tworzenie pierwszej Xamarin.Forms aplikacji



[kolejne aplikacje](#)

[samouczki](#)

[Deep Dive](#)

Colors

MainPage.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="colors.MainPage">
    <StackLayout x:Name="StackLayout">
        <!-- górny element z napisem Colors -->
        <Frame BackgroundColor="#aabbcc" Padding="24" CornerRadius="0">
            <Label Text="Colors" HorizontalTextAlignment="Center" TextColor="White" FontSize="36"/>
        </Frame>
        <!-- przycisk manualnej zmiany koloru -->
        <Button Text="Change color manually" Clicked="BtnManually_Clicked" />
        <!-- przycisk automatycznej zmiany koloru -->
        <Button Text="Change color automatic" Clicked="BtnAutomatic_Clicked" />
        <ListView x:Name="listView" ItemTapped="listView_ItemTapped" ItemsSource="{Binding .}" />
    </StackLayout>
</ContentPage>
```

MainPage.xaml.cs

obsługa przycisku
manualnej zmiany koloru

```
// flaga zmiany przycisku (manualny i automatyczny)
private bool buttonFlag = true;
// pole koloru
private Color color = Color.White;

// obsługa przycisku manualnej zmiany koloru
Odwolania: 0
void BtnManually_Clicked(object sender, System.EventArgs e)
{
    // ustawiamy flagę przycisku - gdy włączona jest automatyczna zmiana koloru
    // nastąpi jej przerwanie
    buttonFlag = false;
    // pobieramy losowy kolor
    color = Color.FromRgb(random.Next(256), random.Next(256), random.Next(256));
    // przypisujemy losowy kolor do komponentu StackLayout
    StackLayout.BackgroundColor = color;
}
```

bindowanie
ListView

MainPage.xaml.cs
class MainPage

obsługa ListView

```
public MainPage()
{
    InitializeComponent();
    // bindowanie ListView - łączenie kolorów z kolejnymi pozycjami ListView (items)
    this.BindingContext = new [] { "red", "green", "blue", "yellow", "AliceBlue", "Aqua" };
}

// obsługa zdarzenia dotknięcia pozycji ListView
Odwolania: 0
private async void listView_ItemTapped(object sender, ItemTappedEventArgs e)
```

Galeria – dodawanie obrazków 1/3

The screenshot shows the Visual Studio IDE with the following components:

- Menu Bar:** Plik, Edycja, Widok, Git, Projekt, Kompilowanie, Debuguj, Test, Analiza, Narzędzia, Rozszerzenia, Okno, Pomoc. Search: Wyszukaj (Ctrl+Q).
- Toolbar:** Debug, Any CPU, galeria.Android, Samsung SM-A750FN (Android 10.0 — API 29).
- Code Editor:** MainPage.xaml.cs, App.xaml.cs, App.xaml. ContentPage XAML code:

```
4 x:Class="galeria.MainPage">
5
6 <StackLayout>
7 <!-- <Frame BackgroundColor="#2196F3" Padding="24" CornerRadius="0">
8 <Label Text="Welcome to Xamarin.Forms!" HorizontalTextAlignment="Center" TextColor="White" FontSize="18" />
9 </Frame-->
10 <Label Text="Galeria" Padding="30,10,30,10" FontSize="Title"/>
11 <Image x:Name="Picture" Source="kot1.png"/>
12
13 </StackLayout>
14
15 </ContentPage>
16
```
- Solution Explorer:** galeria.Android > Resources > drawable. Files: mipmap-anydpi-v26 (icon.xml, icon_round.xml), mipmap-hdpi (icon.png, launcher_foreground.png), mipmap-mdpi (icon.png).
- Properties Window:** drawable Właściwości folderu. Różne: Nazwa folderu: drawable.
- Output Window:** Dane wyjściowe. Debugowanie. Thread finished: <Thread Pool> #14, #13, #9.
- Status Bar:** Gotowe, Dodaj do kontroli źródła, Wybierz repozytorium, 10:30, 26.01.2022.

Galeria – dodawanie obrazków 2/3

The screenshot shows the Visual Studio IDE with the following components:

- Menu Bar:** Plik, Edycja, Widok, Git, Projekt, Kompilowanie, Debuguj, Test, Analiza, Narzędzia, Rozszerzenia, Okno, Pomoc. Search: Wyszukaj (Ctrl+Q).
- Toolbar:** Debug, Any CPU, galeria.Android, Samsung SM-A750FN (Android 10.0 — API 29).
- Code Editor:** MainPage.xaml, MainPage.xaml.cs, App.xaml.cs, App.xaml. ContentPage XAML code:

```
4 x:Class="galeria.MainPage">
5
6 <StackLayout>
7 <!-- <Frame BackgroundColor="#2196F3" Padding="24" CornerRadius="0">
8 <Label Text="Welcome to Xamarin.Forms!" HorizontalTextAlignment="Center" TextColor="White" FontSize="
9 </Frame-->
10 <Label Text="Galeria" Padding="30,10,30,10" FontSize="Title"/>
11 <Image x:Name="Picture" Source="kot1.png"/>
12
13 </StackLayout>
14
15 </ContentPage>
16
```
- Context Menu:** Nowy element... (Ctrl+Shift+A), Istniejący element... (Shift+Alt+A), Nowy folder, Klasa..., Nowa konfiguracja EditorConfig.
- Eksplorator rozwiązań (Solution Explorer):** galeria.Android, Properties, Odwołania, Assets, Resources. Context menu: Dodaj, Dodaj odwołanie sieci Web..., Zawęż zakres do tego elementu, Zmień widok na, Nowy widok — Eksplorator rozwiązań, Wyłącz z projektu, Wytnij (Ctrl+X), Kopiuj (Ctrl+C), Wklej (Ctrl+V), Usuń (Del), Zmień nazwę (F2), Kopiuj pełną ścieżkę, Otwórz folder w Eksploratorze plików, Otwórz w terminalu, Właściwości (Alt+Enter), Różne.
- Output Window:** Dane wyjściowe, Debugowanie. Log messages: Thread finished: <Thread Pool> #14, Wątek 0xe zakończył działanie z kodem 0 (0x0), Thread finished: <Thread Pool> #13, Wątek 0xd zakończył działanie z kodem 0 (0x0), Thread finished: <Thread Pool> #9, Wątek 0x9 zakończył działanie z kodem 0 (0x0).
- Taskbar:** Windows Start button, search bar (Wpisz tu wyszukiwane słowa), taskbar icons, system tray (10:31, 26.01.2022).

Galeria – dodawanie obrazków 3/3

The screenshot shows the following components:

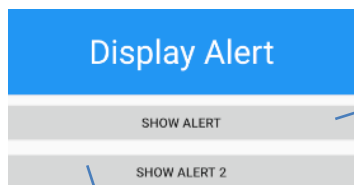
- MainPage.xaml:** XML code for a `StackLayout` containing an `Image` element with `x:Name="Picture"`.
- Dialog:** "Dodaj istniejący element - galeria.Android" with a file explorer showing the path `Dysk USB (F:) > kot`. The file list is as follows:

Nazwa	Data modyfikacji	Typ
kot1	26.01.2022 10:09	Plik PNG
kot2	26.01.2022 10:09	Plik PNG
kot3	26.01.2022 10:09	Plik PNG
kot4	26.01.2022 10:09	Plik PNG

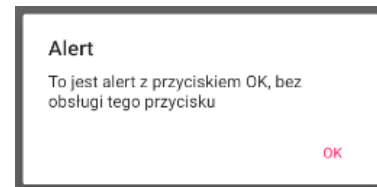
- Project Explorer:** Shows the `drawable` folder under `galeria.Android` with subfolders for different screen densities: `mipmap-anydpi-v26`, `mipmap-hdpi`, and `mipmap-mdpi`.
- Properties Panel:** Shows the properties for the `drawable` folder, including its name and location.
- Output Window:** Displays log messages such as "Thread finished: <Thread Pool> #14" and "Wątek 0xe zakończył działanie z kodem 0 (0x0)".

DisplayAlert

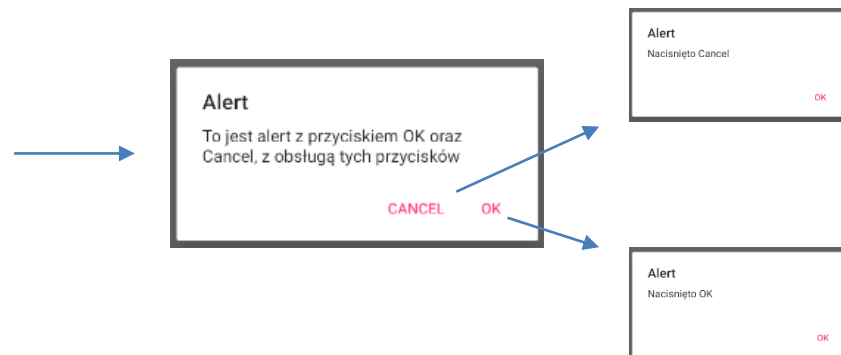
```
<Button Text="show alert" Clicked="Button_Clicked" />  
<Button Text="show alert 2" Clicked="Button_Clicked_1" />
```



```
private async void Button_Clicked(object sender, EventArgs e)  
{  
    await DisplayAlert("Alert", "To jest alert z przyciskiem OK, bez obsługi tego przycisku", "OK");  
}
```

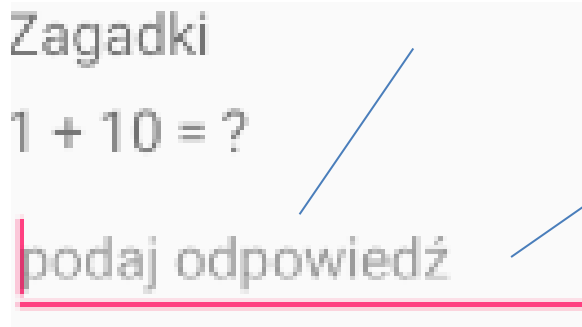


```
private async void Button_Clicked_1(object sender, EventArgs e)  
{  
    var odp = await DisplayAlert("Alert",  
        "To jest alert z przyciskiem OK oraz Cancel, z obsługą tych przycisków",  
        "OK",  
        "Cancel");  
  
    if (odp == true)  
    {  
        await DisplayAlert("Alert", "Nacisnięto OK", "OK");  
    }  
    else  
    {  
        await DisplayAlert("Alert", "Nacisnięto Cancel", "OK");  
    }  
}
```



Entry

entryOdpowiedz.Text



Entry (okienko input)

```
if (entryOdpowiedz.Text == 11){  
    // ok  
}
```

```
<!-- ustawiamy klawiaturę numeryczną, zdarzenie Completed zachodzi gdy użytkownik wpisze liczbę i naciśnie OK -->  
<Entry x:Name="entryOdpowiedz" Placeholder="podaj odpowiedź" Keyboard="Numeric" Completed="entryOdpowiedz_Completed"/>
```

Nazwa komponentu Entry

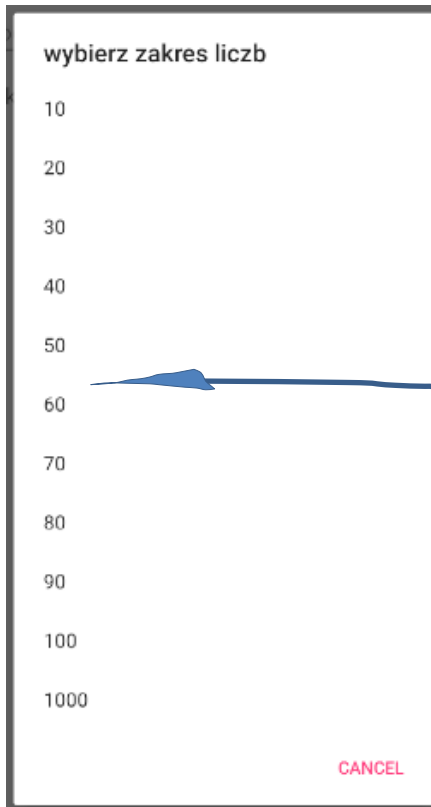
Ustawienie numerycznej klawiatury

Placeholder

Funkcja obsługi zdarzenia komponentu Entry

```
// metoda obsługująca pole textowe po wpisaniu wyniku i naciśnięciu ok  
Odwwołania: 0  
private void entryOdpowiedz_Completed(object sender, EventArgs e)  
{
```

Picker



Picker

Binding

SelectedIndexChanged handling

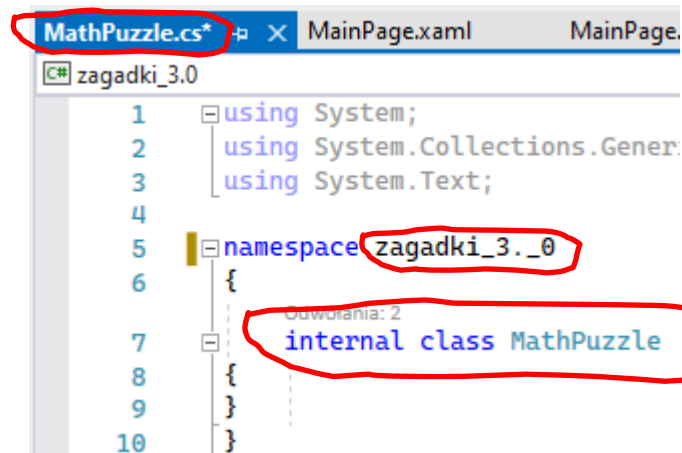
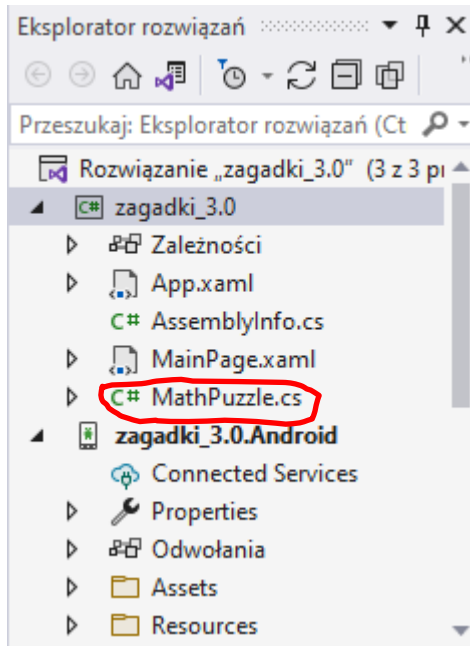
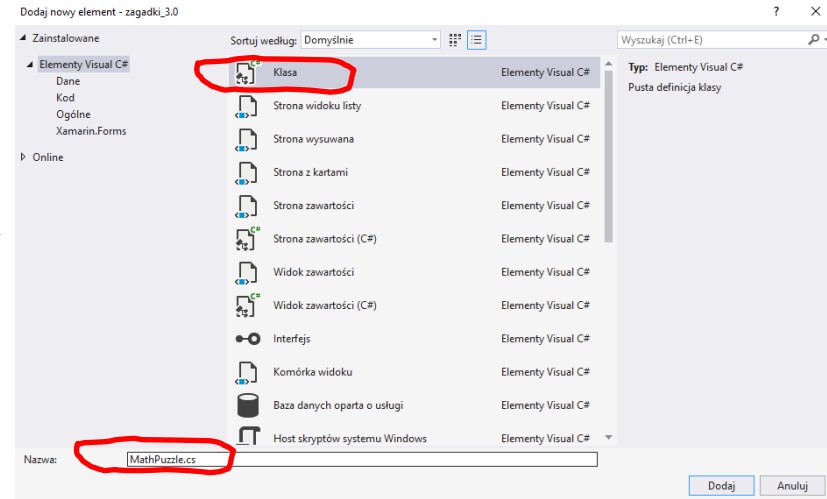
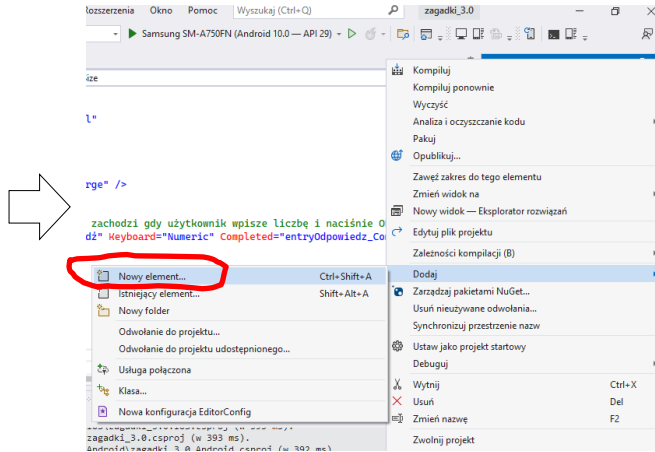
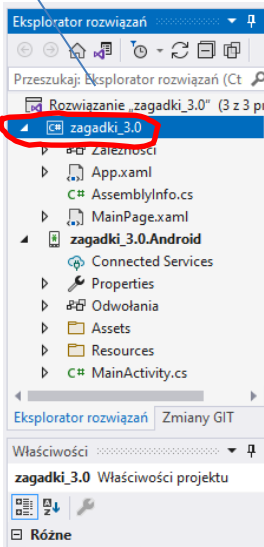
```
<!-- lista wyboru zakresu liczb -->  
<StackLayout Orientation="Horizontal">  
  <Label x:Name="labelZakres"  
    Text="zakres"  
    FontSize="Medium"  
    VerticalTextAlignment="Center"/>  
  <Picker x:Name="pickerZakres"  
    Title="wybierz zakres liczb"  
    ItemsSource="{Binding .}"  
    SelectedIndexChanged="pickerZakres_SelectedIndexChanged" />  
</StackLayout>
```

```
public MainPage()  
{  
  InitializeComponent();  
  labelPytanie.Text = generateCalculation();  
  // bindujemy pickerZakres ItemsSource  
  // lista przedstawia zakresy liczb w równaniach  
  this.BindingContext = new[] { 10,20,30,40,50,60,70,80,90,100,1000 };  
}
```

```
// metoda wybierająca zakres liczb za pomocą komponentu Picker  
Odwolania: 0  
void pickerZakres_SelectedIndexChanged(object sender, EventArgs e)  
{  
  var picker = (Picker)sender;  
  int selectedIndex = picker.SelectedIndex;  
  
  if (selectedIndex != -1)  
  {  
    // wybieramy zakres liczb  
    range = (int)picker.ItemsSource[selectedIndex];  
    // wybieramy pytanie z nowym zakresem  
    labelPytanie.Text = generateCalculation();  
  }  
}
```

Dodawanie nowej klasy MathPuzzle do projektu zagadki_3.0

Prawy myszy



Data i czas

```
public MainPage()  
{  
    InitializeComponent();  
    labelData.Text = DateTime.Now.ToString("dddd, dd MMMM yyyy, HH:mm:ss");  
}
```



Friday, 02 September 2022,
16:53:36

czas stoi w miejscu

[Standardowe ciągi formatujące datę i godzinę](#)
[przykłady daty i godziny](#)

Data i czas

```
labelData.Text = datePickerSet.Date.ToString("MM");
```

...tuje miesiąc jako liczbę z zakresu od 01 do 12 (lub od 01 do 13 dla kalendarzy z 13 miesiącami). Miesiąc jednocyfrowy jest

MM	miesiąc (2 cyfry)
mm	minuta (2 cyfry)
MMM	miesiąc (skrót)
MMMM	miesiąc (pełny)

```
daySet = datePickerSet.Date.ToString("dd");
```

```
monthSet = datePickerSet.Date.ToString("ddd");
```

```
yearSet = datePickerSet.Date.ToString("ddddd");
```

dd	dzień miesiąca (2 cyfry)
ddd	dzień tygodnia (skrót)
dddd	dzień tygodnia (pełna nazwa)

Przykłady:
dd (pl-PL) → 15
dd (en-US) → 15

Ciąg formatu niestandardowego „dd” reprezentuje dzień miesiąca jako liczbę z zakresu od 01 do 31. Dzień 1-cyfrowy jest wyświetlany w formacie z wiodącym zerem.

```
yearSet = datePickerSet.Date.ToString("yyyy");
```

```
labelData.Text = datePickerSet.Date.ToString("yyyy-MM-dd");
```

yyyy	rok (4 cyfry)
yyyyy	rok (5 cyfr)

Przykłady:
yyyy (pl-PL) → 2009
yyyy (en-US) → 2009

Niestandardowy specyfikator formatu „yyyy” reprezentuje rok za pomocą minimalnie 4 cyfr. Jeśli rok ma więcej niż cztery cyfry znaczące, są one uwzględniane w ciągu wyniku. Jeśli rok ma mniej niż cztery cyfry, liczba jest uzupełniana zerami wiodącymi, tak aby otrzymać cztery cyfry.

Data i czas

```
public MainPage()
{
    InitializeComponent();
    // uruchomienie timera, który
    // co 1s uruchamia metodę OnTimerTick
    Device.StartTimer(TimeSpan.FromSeconds(1), OnTimerTick);
}
```



```
bool OnTimerTick()
{
    labelData.Text = DateTime.Now.ToString("dddd, dd MMMM yyyy, HH:mm:ss");
    return true;
}
```



Friday, 02 September 2022,
16:53:36

czas płynie

Data i czas

zadanie:

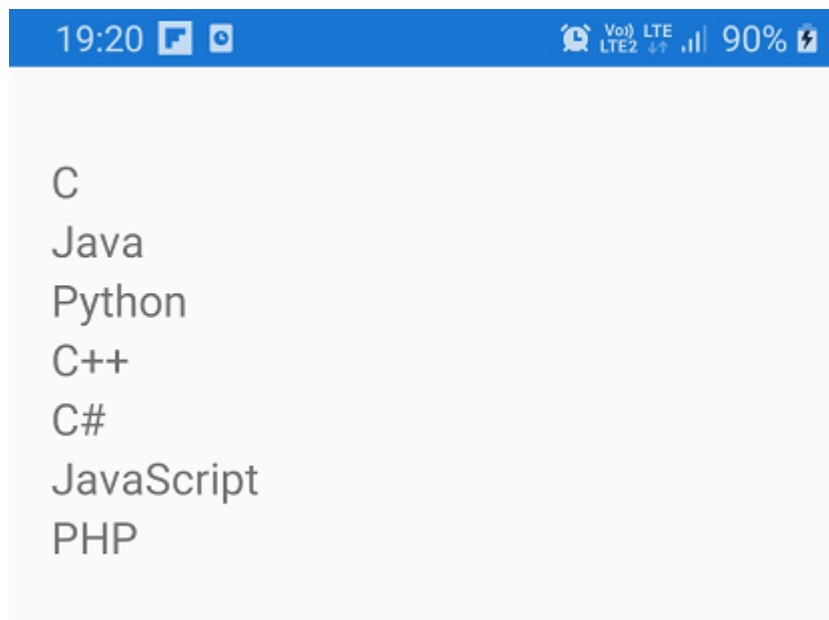
- napisać program pokazujący aktualną datę, której format zmienia się co 3 sekundy (program działa w kółko)

CollectionView programmingLanguages

```
<?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="programmingLanguages.MainPage">
  <StackLayout Margin="20,35,20,20">
    <CollectionView>
      <CollectionView.ItemsSource>
        <x:Array Type="{x:Type x:String}">
          <x:String>C</x:String>
          <x:String>Java</x:String>
          <x:String>Python</x:String>
          <x:String>C++</x:String>
          <x:String>C#</x:String>
          <x:String>JavaScript</x:String>
          <x:String>PHP</x:String>
        </x:Array>
      </CollectionView.ItemsSource>
    </CollectionView>
  </StackLayout>
</ContentPage>
```

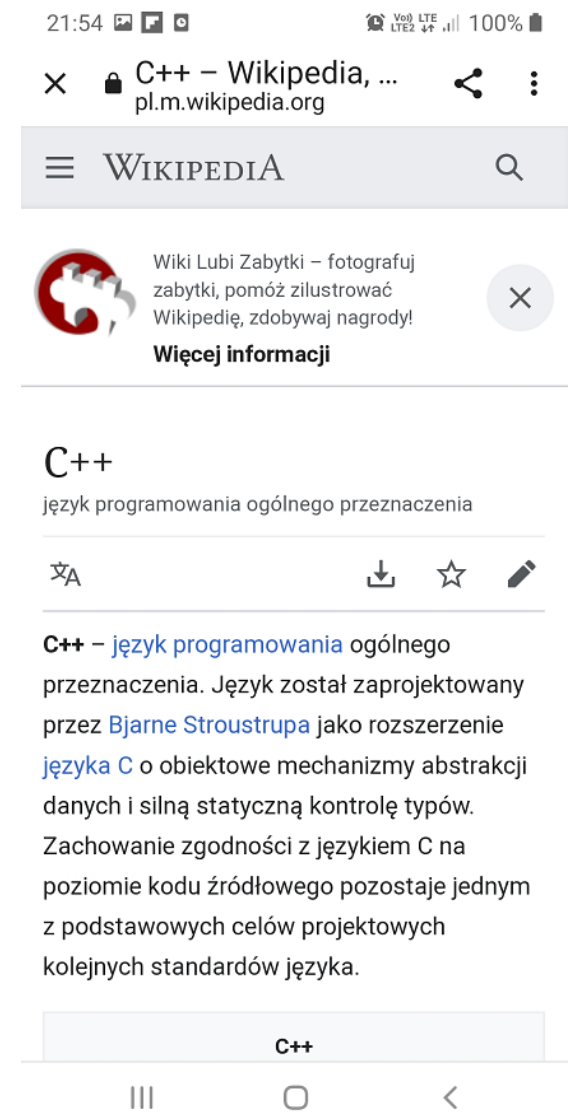
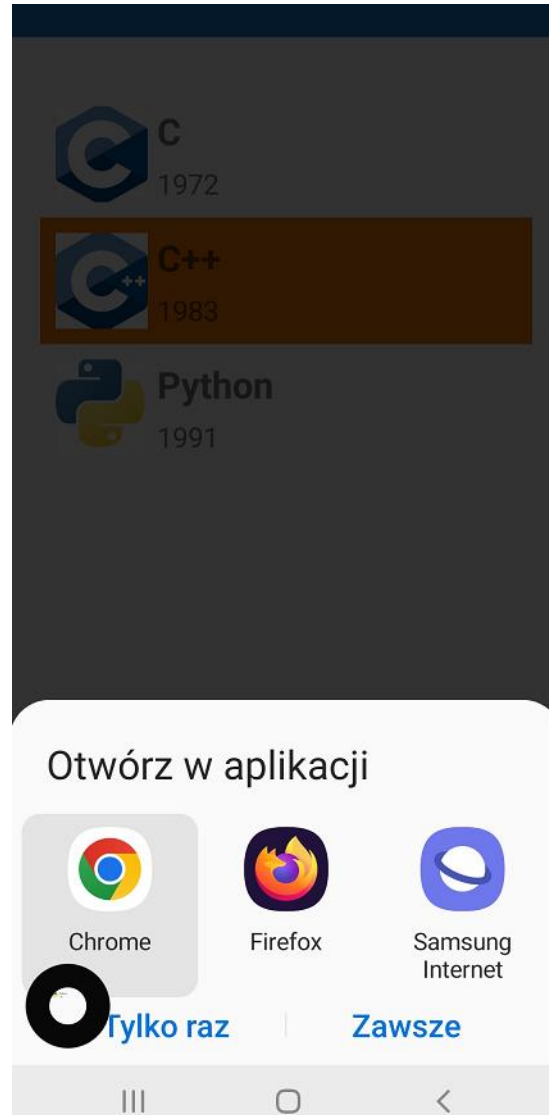
MainPage.xaml

wypełnianie CollectionView w pliku xaml



[Tworzenie obiektu CollectionView](#)

CollectionView programmingLanguages1



CollectionView

```
<?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="programmingLanguages1.MainPage">
  <StackLayout Margin="20,35,20,20">
    <CollectionView ItemsSource="{Binding Languages}"
      SelectionMode="Single"
      SelectionChanged="OnSelectionChanged">
      <CollectionView.ItemTemplate>
        <DataTemplate>
          <Grid Padding="10"
            RowDefinitions="Auto, *"
            ColumnDefinitions="Auto, *">
            <Image Grid.RowSpan="2"
              Source="{Binding Image}"
              Aspect="AspectFill"
              HeightRequest="60"
              WidthRequest="60" />
            <Label Grid.Column="1"
              Text="{Binding Name}"
              FontAttributes="Bold" />
            <Label Grid.Row="1"
              Grid.Column="1"
              Text="{Binding Year}"
              VerticalOptions="End" />
          </Grid>
        </DataTemplate>
      </CollectionView.ItemTemplate>
    </CollectionView>
  </StackLayout>
</ContentPage>
```

binding (wiązanie danych)



// klasa języka

Odwołania: 7

public class Language

{

Odwołania: 4

public string Name { get; set; }

Odwołania: 3

public string Year { get; set; }

Odwołania: 4

public string Url { get; set; }

Odwołania: 3

public string Image { get; set; }

Odwołania: 0

public override string ToString()

{

return Name;

}

}

MainPage.xaml

nowa klasa Language.cs

[Tworzenie obiektu CollectionView](#)

CollectionView

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;
using Xamarin.Essentials; // przeglądarka
```

```
namespace programmingLanguages1
```

```
{
```

```
    Odwołania: 5
```

```
    public partial class MainPage : ContentPage
```

```
    {
```

```
        // lista języków
```

```
        Odwołania: 4
```

```
        public IList<Language> Languages { get; private set; }
```

```
        // konstruktor
```

```
        1 odwołanie
```

```
        public MainPage()
```

```
        {
```

```
            InitializeComponent();
```

```
            // zainicjalizowanie listy języków
```

```
            Languages = new List<Language>();
```

```
            // wypełnienie listy języków poszczególnymi językami
```

```
            Languages.Add(new Language
```

```
            {
```

```
                Name = "C",
```

```
                Year = "1972",
```

```
                Image = "C_Programming_Language.png",
```

```
                Url = "https://en.wikipedia.org/wiki/C_(programming_language)"
```

```
            });
```

```
            Languages.Add(new Language
```

```
            {
```

```
                Name = "C++",
```

```
                Year = "1983",
```

```
                Image = "Cplusplus.png",
```

```
                Url = "https://pl.wikipedia.org/wiki/C%2B%2B"
```

```
            });
```

```
            Languages.Add(new Language
```

```
            {
```

```
                Name = "Python",
```

```
                Year = "1991",
```

```
                Image = "python.png",
```

```
                Url = "https://en.wikipedia.org/wiki/Python_%28programming_language%29"
```

```
            });
```

```
            // bindowanie
```

```
            BindingContext = this;
```

```
        }
```

MainPage.xaml.cs 1/2

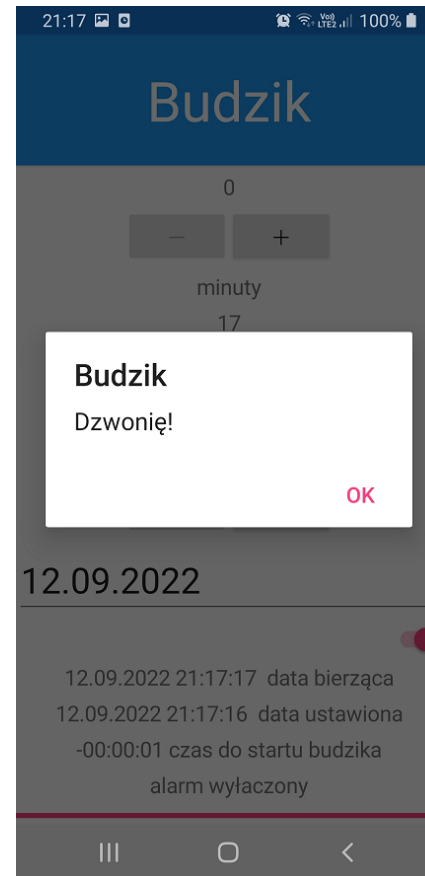
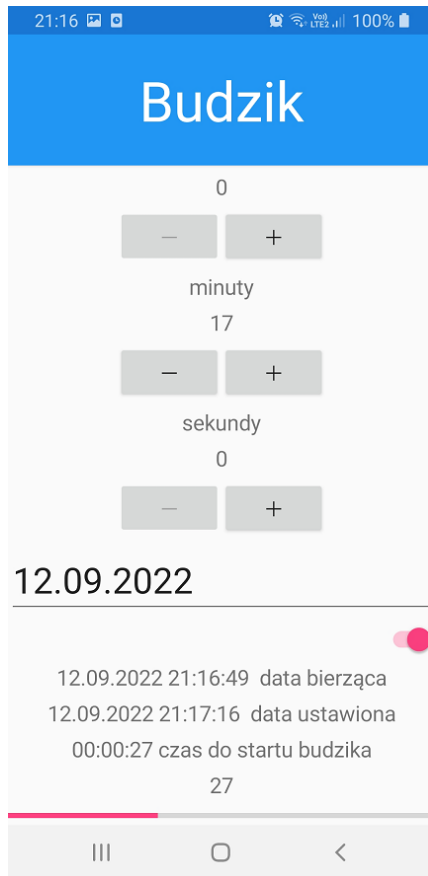
CollectionView

```
// metoda trigerowana przez wybranie elementu CollectionView
Odwołania: 0
void OnSelectionChanged(object sender, SelectionChangedEventArgs e)
{
    // uaktywniamy wybrany element (zmienia się tło)
    Language selectedItem = e.CurrentSelection[0] as Language;
    // otwieramy przeglądarkę, url
    openBr(new Uri(selectedItem.Url));
}

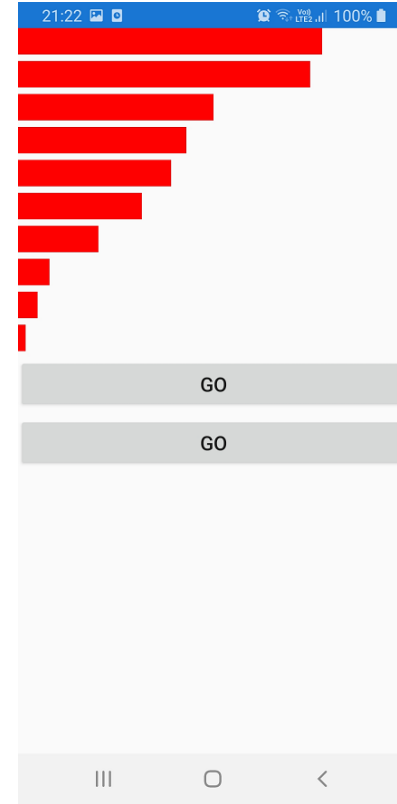
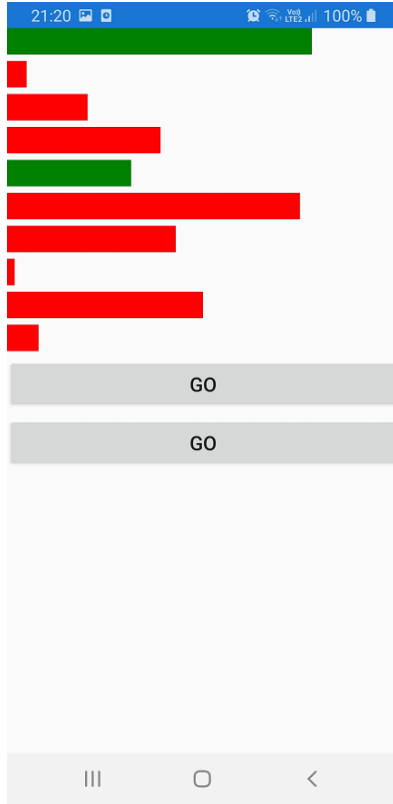
// metoda asynchroniczna otwierająca przeglądarkę
1 odwołanie
public async void openBr(Uri uri)
{
    await Browser.OpenAsync(uri, BrowserLaunchMode.SystemPreferred);
}
}
```

MainPage.xaml.cs 2/2

timer



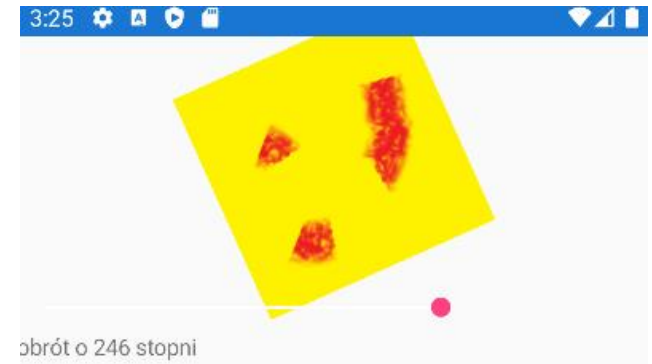
selectionSort



binding



wiązanie suwaka
do obrotu zdjęcia



```
<StackLayout>
```

```
  <Image Source="yellowBox.png"  
    BindingContext="{x:Reference Name=slider}"  
    Rotation="{Binding Path=Value}" />
```

Image Rotation wiążemy
z Slider Value

```
  <Slider x:Name="slider"  
    Maximum="360" />
```

Image wiążemy ze Sliderem

```
  <Label BindingContext="{x:Reference slider}"  
    Text="{Binding Value, StringFormat='obróć o {0:F0} stopni'}" />
```

```
</StackLayout>
```

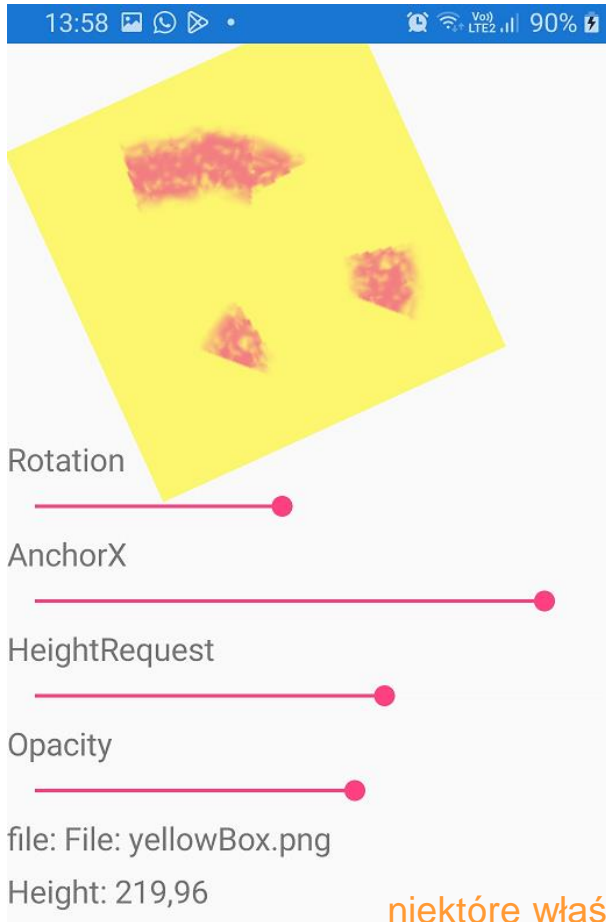
Label Text wiążemy
z Slider Value, narzucamy określony format tekstu

liczba całkowita
{0:F2} – liczba z 2 wartościami
po przecinku

binding ścieżka powiązania

binding

wiążemy wiele suwaków do jednego zdjęcia



niektóre właściwości
można powiązać

image attributes

```
<StackLayout>
  <Image x:Name="image"
        Source="yellowBox.png" IsOpaque="True"/>

  <Label Text="Rotation" />
  <Slider x:Name="slider"
        BindingContext="{x:Reference Name=image}"
        Value="{Binding Path=Rotation, Mode=OneWayToSource}"
        Maximum="360" />

  <Label Text="AnchorX" />
  <Slider x:Name="slider1"
        BindingContext="{x:Reference Name=image}"
        Value="{Binding Path=AnchorX, Mode=TwoWay}"
        Maximum="0.5" />

  <Label Text="HeightRequest" />
  <Slider x:Name="slider2"
        BindingContext="{x:Reference Name=image}"
        Value="{Binding Path=HeightRequest, Mode=TwoWay}"
        Maximum="360" />

  <Label Text="Opacity" />
  <Slider x:Name="slide3"
        BindingContext="{x:Reference Name=image}"
        Value="{Binding Path=Opacity, Mode=TwoWay}"
        Maximum="1" />

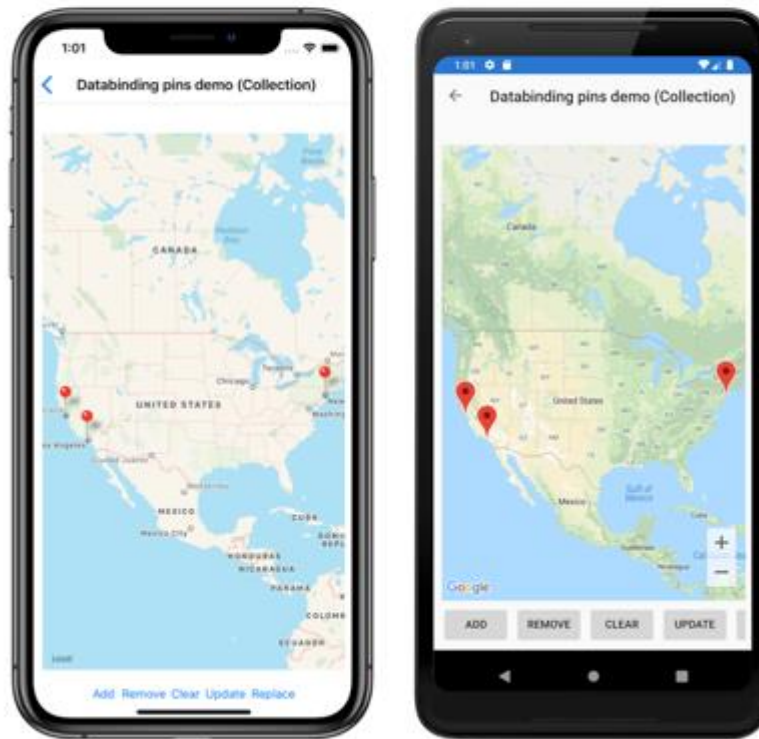
  <Label BindingContext="{x:Reference image}"
        Text="{Binding Source, StringFormat='file: {0}}'" />
  <Label BindingContext="{x:Reference image}"
        Text="{Binding Height, StringFormat='Height: {0}}'" />
</StackLayout>
```

mapy

Working with maps in Xamarin.Forms

The `Map` control displays a map, and requires the `Xamarin.Forms.Maps` NuGet package.

For more information, see [Xamarin.Forms Map](#).



[github.com WorkingWithMaps](https://github.com/WorkingWithMaps)

mapy konfiguracja

Android

Proces konfiguracji wyświetlania i interakcji z mapą w systemie Android to:

1. Pobierz klucz interfejsu API Mapy Google i dodaj go do manifestu.
2. Określ numer wersji usług Google Play w manifeście.
3. Określ wymaganie dla biblioteki Apache HTTP Legacy w manifeście.
4. [opcjonalnie] Określ uprawnienie `WRITE_EXTERNAL_STORAGE` w manifeście.
5. [opcjonalnie] Określ uprawnienia lokalizacji w manifeście.
6. [opcjonalnie] Załadaj uprawnienia lokalizacji środowiska uruchomieniowego `MainActivity` w klasie.

wyświetlenie lokalizacji telefonu mapie

[mapy](#)

Menadżer pakietów NuGet

The screenshot shows the Visual Studio interface with the NuGet Package Manager window open. The menu 'Projekt' is open, and 'Zarządzaj pakietami NuGet...' is selected. The Package Manager console on the right shows a list of installed packages.

Package Name	Version	Download Size
AutoMapper.Ext	11.0.0	111M
System.IO.MemoryStream	4.3.0	
Mapster	7.3.0	
Mapster.Core	1.2.0	
StructureMap	4.7.1	22.1M
Xamarin.Forms.Maps	5.0.0.2515	6.2M
Abp.AutoMapper	7.3.0	4.5M

Menadżer pakietów NuGet

The screenshot shows the Visual Studio IDE with the NuGet Package Manager window open. The main window displays a list of packages, and the right-hand pane shows the details for the selected package, 'Xamarin.Forms.Maps'.

Przełączniki: Przeglądaj, Zainstalowane, Aktualizacje 2

Wyszukiwanie: map

Źródło pakietów: nuget.org

Nazwa pakietu	Wersja
AutoMapper.Extensions.Microsoft.DependencyInjection przez: Jimmy Bogard, Pobrania: 111M	11.0.0
System.IO.MemoryMappedFiles przez: Microsoft, Pobrania: 72,1M	4.3.0
Mapster przez: chaowlert, eric_swann, andrerav, Pobrania: 7,55M	7.3.0
Mapster.Core przez: Chaowlert Chaisrichalernpol, Pobrania: 4,06M	1.2.0
StructureMap przez: Jeremy D. Miller, Joshua Flanagan, Frank Quednau, Dmytro Dzumia, Pobrania: 22,1M	4.7.1
Xamarin.Forms.Maps przez: Microsoft, Pobrania: 6,2M	5.0.0.2515
Abp.AutoMapper przez: Abp.AutoMapper, Pobrania: 4,5M	7.3.0
System.Security.Principal.Windows przez: Microsoft, Pobrania: 1,21B	5.0.0

Xamarin.Forms.Maps (nuget.org)

Wersja: Najnowsza stabilna 5.0.0.2515 **Zainstaluj**

Opcje

Opis
Maps models and renderers for Xamarin.Forms

Wersja: 5.0.0.2515+946-sha.94acebbb4-azdo.6439792

Autorzy: Microsoft

Licencja: MIT

Data publikacji: poniedziałek, 25 lipca 2022 (25.07.2022)

Adres URL projektu: <http://xamarin.com/forms>

Zgłoś nadużycie: <https://www.nuget.org/packages/Xamarin.Forms.Maps/5.0.0.2515/ReportAbuse>

Tagi: xamarin, forms, maps, xamarinforms, xamarinformsmaps, xamarin.forms.maps

Zależności

- Any,Version=v0.0
 - Xamarin.Forms (>= 5.0.0.2515)
- MonoAndroid,Version=v10.0
 - Xamarin.GooglePlayServices.Maps (>= 117.0.0)
 - Xamarin.Forms (>= 5.0.0.2515)
- Tizen,Version=v4.0
 - Tizen.NET (>= 4.0.0)

Menadżer pakietów NuGet

Menadżer pakietów NuGet: mapy

Źródło pakietów: nuget.org

Xamarin.Forms

Zainstalowane: 5.0.0.2196
Wersja: Najnowsza stabilna 5.0.0.2515

Opis
Build native UIs for iOS, Android, UWP, macOS, Tizen and many more from a single, shared C# codebase

Wersja: 5.0.0.2515+946-sha.94acebbb4-azdo.6439792
Autorzy: Microsoft
Licencja: MIT
Data publikacji: poniedziałek, 25 lipca 2022 (25.07.2022)
Adres URL projektu: <http://xamarin.com/forms>
Zgłoś nadużycie: <https://www.nuget.org/packages/Xamarin.Forms/5.0.0.2515/ReportAbuse>
Tagi: xamarin, forms, xamarinforms, xamarin.forms

Zależności

- MonoAndroid,Version=v10.0
 - Xamarin.AndroidX.Lifecycle.LiveData (>= 2.3.1.1 && < 2.4.0)
 - Xamarin.Google.Android.Material (>= 1.4.0.2 && < 1.5.0)
 - Xamarin.AndroidX.Legacy.Support.V4 (>= 1.0.0.8 && < 1.1.0.0)
 - Xamarin.AndroidX.Browser (>= 1.3.0.6 && < 1.4.0)
 - Xamarin.AndroidX.Core (>= 1.6.0.1 && <= 1.7.0.2)
 - Xamarin.AndroidX.CustomView (>= 1.1.0.7 && < 1.2.0)
 - Xamarin.AndroidX.Preference (>= 1.1.1.0.8 && < 1.2.0)

Lista błędów

Całe rozwiązanie	2 Błędy	0 Ostrzeżenia	0 Komunikaty	Kompilacja + IntelliSense	Przeszukaj listę błędów
Kod	Opis	Projekt	Plik	W...	Stan pominięcia
	Przywracanie pakietów nie powiodło się. Wycyfywanie zmian pakietów dla „mapy”.				
NU1605	Wykryto obniżenie wersji pakietu: Xamarin.Forms z 5.0.0.2515 na 5.0.0.2196. Aby wybrać inną wersję, odwołaj się do pakietu bezpośrednio z projektu. mapy -> Xamarin.Forms.Maps 5.0.0.2515 -> Xamarin.Forms (>= 5.0.0.2515) mapy -> Xamarin.Forms (>= 5.0.0.2196)	mapy	mapy.csproj		1

Menadżer pakietów NuGet

The screenshot shows the NuGet Package Manager interface in Visual Studio. The main window displays the details for the **Xamarin.Forms.Maps** package, version 5.0.0.2515. The package is provided by Microsoft and is used for Maps models and renderers for Xamarin.Forms. The interface includes a search bar, a list of installed packages, and a detailed view of the selected package. The detailed view shows the package's metadata, including its version, author, license, publication date, project URL, and a list of dependencies. The dependencies are: Any, Version=v0.0; Xamarin.Forms (>= 5.0.0.2515); MonoAndroid, Version=v10.0; Xamarin.GooglePlayServices.Maps (>= 117.0.0); Xamarin.Forms (>= 5.0.0.2515); and Tizen, Version=v4.0; Tizen.NET (>= 4.0.0).

Przeładowanie | **Zainstalowane** | Aktualizacje

maps Uwzględnij wersję wstępną

Xamarin.Forms.Maps przez: Microsoft
Maps models and renderers for Xamarin.Forms

5.0.0.2515

Xamarin.Forms.Maps

Zainstalowane: 5.0.0.2515

Wersja: 5.0.0.2515

Opcje

Opis
Maps models and renderers for Xamarin.Forms

Wersja: 5.0.0.2515+946-sha.94acebbb4-azdo.6439792

Autorzy: Microsoft

Licencja: MIT

Data publikacji: poniedziałek, 25 lipca 2022 (25.07.2022)

Adres URL projektu: <http://xamarin.com/forms>

Zgłoś nadużycie: <https://www.nuget.org/packages/Xamarin.Forms.Maps/5.0.0.2515/ReportAbuse>

Tagi: xamarin, forms, maps, xamarinforms, xamarinformsmaps, xamarin.forms.maps

Zależności

- Any, Version=v0.0
 - Xamarin.Forms (>= 5.0.0.2515)
- MonoAndroid, Version=v10.0
 - Xamarin.GooglePlayServices.Maps (>= 117.0.0)
- Xamarin.Forms (>= 5.0.0.2515)
- Tizen, Version=v4.0
 - Tizen.NET (>= 4.0.0)

Licencja na każdy pakiet jest udzielana przez jego właściciela. Firma NuGet nie ponosi odpowiedzialności za pakiety innych firm ani nie udziela na nie żadnych licencji.

Do not show this again

Lista błędów | Całe rozwiązanie | | Przeszukaj listę błędów

Kod	Opis	Projekt	Plik	W...	Stan pominięcia
-----	------	---------	------	------	-----------------

Obtaining a Google Maps API Key

To use the Google Maps functionality in Android, you need to register for a Maps API key with Google. Until you do this, you will just see a blank grid instead of a map in your applications. You must obtain a Google Maps Android API v2 key - keys from the older Google Maps Android API key v1 will not work.

<https://learn.microsoft.com/pl-pl/xamarin/android/platform/maps-and-location/maps/obtaining-a-google-maps-api-key?tabs=windows>

Obtaining a Google Maps API Key



Pierwsze kroki w Google Maps Platform

Możesz teraz zacząć programować. To jest klucz interfejsu API, którego potrzebujesz do wdrożenia. Klucz interfejsu API można znaleźć w sekcji Dane logowania.

Twój klucz interfejsu API

dfg65y56umfmmymr6mKHKLKHH67555758568575868



- Włącz wszystkie interfejsy API Map Google w tym projekcie 
- Twórz alerty dotyczące budżetu, które pomogą mi zapanować nad wydatkami i przypomną o tym, że zbliżam się do miesięcznego limitu środków na Mapy Google w wysokości 200 USD 

OTWÓRZ GOOGLE MAPS PLATFORM

[Google Developers Console API & Services Dashboard](#)

Obtaining a Google Maps API Key

The screenshot shows a dialog box titled "Chroń swój klucz interfejsu API" (Protect your API key). The text inside reads: "Jeśli chcesz zabezpieczyć swoje konto przed niepożądanym i nieupoważnionym wykorzystaniem, zalecamy ograniczenie klucza interfejsu Google Maps Platform API." (If you want to protect your account from unwanted and unauthorized use, we recommend restricting your Google Maps Platform API key). Below the text is a dropdown menu labeled "Select restriction type" with a downward arrow. At the bottom of the dialog are two buttons: "MOŻE PÓŹNIEJ" (Maybe later) and "OGRANICZ KLUCZ" (Restrict key). A tooltip at the bottom of the dialog says: "Klucz interfejsu API możesz ograniczyć w każdej chwili. Wystarczy, że z nawigacji po sekcji przejdziesz na stronę [Dane logowania](#) i wybierzesz ten klucz." (You can restrict your API key at any time. It's enough that from the navigation in the section you go to the [Login data](#) page and select this key).

Strony odsyłające HTTP (witryny internetowe)

Adresy IP (serwery WWW, zadania cron itp.)

Aplikacje na Androida

Aplikacje na iOS

Ograniczenie interfejsu API

Obtaining a Google Maps API Key

Google Cloud AndroidMapy api key Szukaj

API Interfejsy API i usługi Dane logowania + UTWÓRZ DANE LOGOWANIA USUŃ

Włączone interfejsy API i us...
Biblioteka
Dane logowania
Ekran zgody OAuth
Weryfikacja domeny
Zgody dotyczące użytkow...

Utwórz dane logowania, aby uzyskać dostęp do aktywnych interfejsów API. [Dowiedz się więcej](#)

⚠ Pamiętaj, aby skonfigurować ekran zgody OAuth tak, by zawierał informacje o Twojej aplikacji. [SKONFIGURUJ EKRAŃ ZGODY](#)

Klucze API

<input type="checkbox"/>	Nazwa	Data utworzenia ↓	Ograniczenia	Działania
<input type="checkbox"/>	✓ Maps API Key	6 paź 2022	Maps SDK for Android	POKAŻ KLUCZ

Identyfikatory klienta OAuth 2.0

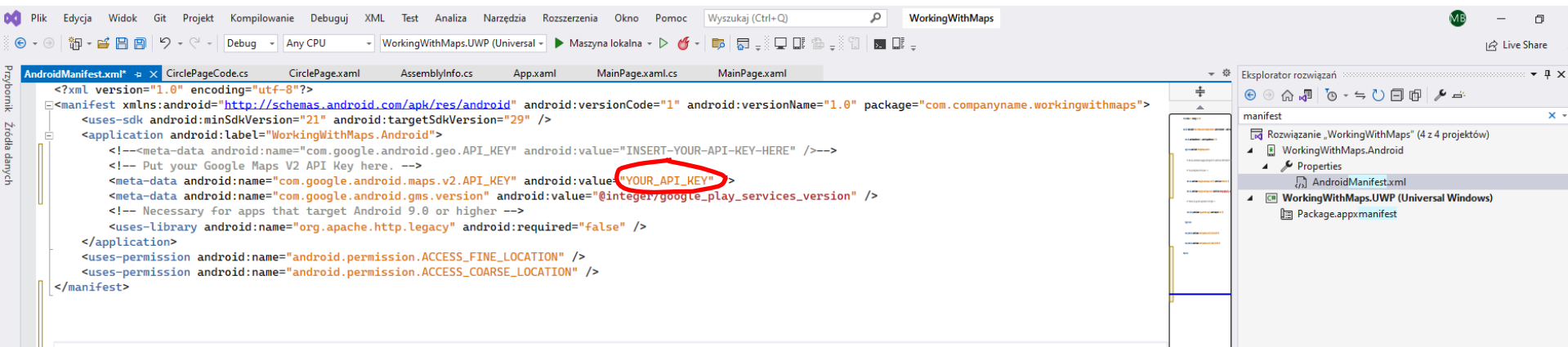
<input type="checkbox"/>	Nazwa	Data utworzenia ↓	Typ	Identyfikator klienta	Działania
Brak klientów OAuth do wyświetlenia					

Konta usługi

<input type="checkbox"/>	E-mail	Nazwa ↑	Działania
Brak kont usługi do wyświetlenia			

[Zarządzaj kontami usługi](#)

Umieszczenie YOUR_API_KEY w pliku Manifest

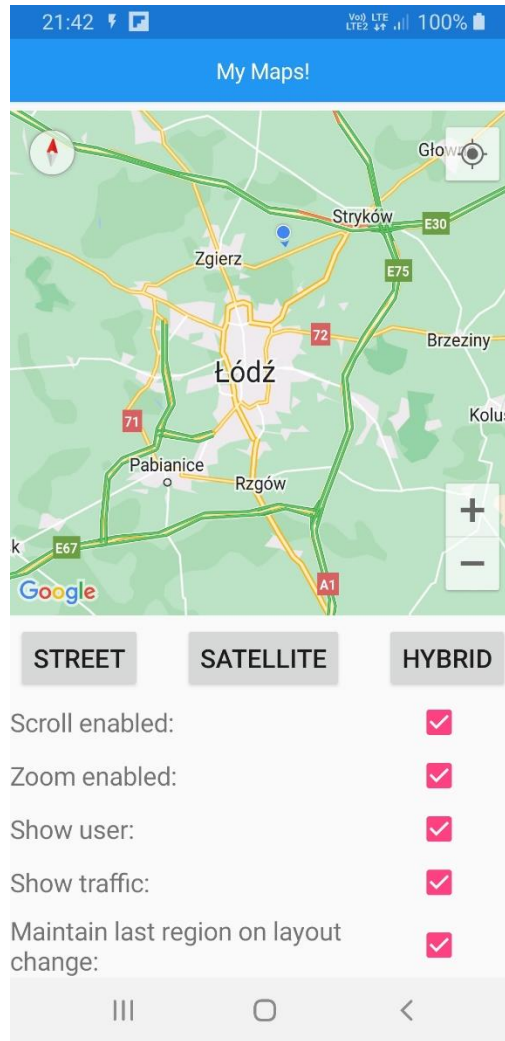


The screenshot displays the Visual Studio IDE with the AndroidManifest.xml file open. The code is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="1" android:versionName="1.0" package="com.companyname.workingwithmaps">
  <uses-sdk android:minSdkVersion="21" android:targetSdkVersion="29" />
  <application android:label="WorkingWithMaps.Android">
    <!--<meta-data android:name="com.google.android.geo.API_KEY" android:value="INSERT-YOUR-API-KEY-HERE" />-->
    <!-- Put your Google Maps V2 API Key here. -->
    <meta-data android:name="com.google.android.maps.v2.API_KEY" android:value="YOUR_API_KEY" />
    <meta-data android:name="com.google.android.gms.version" android:value="@integer/google_play_services_version" />
    <!-- Necessary for apps that target Android 9.0 or higher -->
    <uses-library android:name="org.apache.http.legacy" android:required="false" />
  </application>
  <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
  <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
</manifest>
```

The text "YOUR_API_KEY" in the meta-data tag is circled in red. The Solution Explorer on the right shows the project structure, including the manifest file.

My Maps



Lokalna baza danych

20:57

Enter name

Enter age

ADD TO DATABASE

Jan

15

Katarzyna

7

[Xamarin.Forms Samouczek dotyczący lokalnej bazy danych](#)

Menadżer pakietów NuGet

The screenshot shows the Visual Studio interface with the NuGet Package Manager menu open. The menu options are:

- Dodaj klasę...
- Dodaj nowe źródło danych...
- Dodaj nowy element... (Ctrl+Shift+A)
- Dodaj istniejący element... (Shift+Alt+A)
- Edytuj plik projektu
- Nowy folder
- Pokaż wszystkie pliki
- Zwolnij projekt
- Dodaj odwołanie do projektu...
- Dodaj odwołanie do projektu udostępnionego...
- Dodaj odwołanie do usługi...
- Usługi połączone
- Ustaw jako projekt startowy
- Zależności projektu...
- Kolejność kompilowania projektu...
- Eksportuj szablon...
- Zarządzaj pakietami NuGet...** (highlighted)
- Właściwości binding
- Ustawienia obszaru roboczego narzędzia CMake

The background shows the NuGet Package Manager window with a search for 'binding'. The list of packages includes:

Package Name	Version	Downloads
AutoMapper.Ext	11.0.0	111M
System.IO.MemoryStream	4.3.0	
Mapster	7.3.0	
Mapster.Core	1.2.0	
StructureMap	4.7.1	22,1M
Xamarin.Forms.Maps	5.0.0.2515	6,2M
Abp.AutoMapper	7.3.0	4,5M

Menadżer pakietów NuGet

Dodawanie biblioteki SQLite.NET

The screenshot shows the Visual Studio IDE with the NuGet Package Manager window open. The package 'sqlite-net-pcl' is selected in the list on the left. The right pane displays the details for this package, including its version (1.8.116), description, and dependencies. The bottom pane shows the 'Lista błędów' (Error List) with 0 errors and 0 warnings.

Przebieg: Dodanie biblioteki SQLite.NET do projektu.

Wersja: 1.8.116

Opis: SQLite-net is an open source and light weight library providing easy SQLite database storage for .NET, Mono, and Xamarin applications. This version uses SQLitePCLRaw to provide platform independent versions of SQLite.

Wersja: 1.8.116

Autorzy: SQLite-net

Licencja: Wyświetl licencje

Data publikacji: czwartek, 2 września 2021 (02.09.2021)

Adres URL projektu: <https://github.com/praeclarum/sqlite-net>

Zgłoś nadużycie: <https://www.nuget.org/packages/sqlite-net-pcl/1.8.116/ReportAbuse>

Tagi: sqlite-net, sqlite, database, orm

Zależności:

- .NETStandard,Version=v2.0
- SQLitePCLRaw.bundle_green (>= 2.0.4)

Lista błędów: 0 Błędy, 0 Ostrzeżenia, 0 Komunikaty

Przebieg: Dodaj do kontroli źródła

Cykl życia aplikacji

App.xaml.cs

The image shows a Visual Studio IDE with the following components:

- Code Editor:** Displays the `App.xaml.cs` file with the following code:

```
namespace ApplicationLifecycle
{
    Odwołania: 5
    public partial class App : Application
    {
        1 odwołanie
        public App()
        {
            InitializeComponent();
            MainPage = new MainPage();
        }

        Odwołania: 0
        protected override void OnStart()
        {
            Console.WriteLine("OnStart");
        }

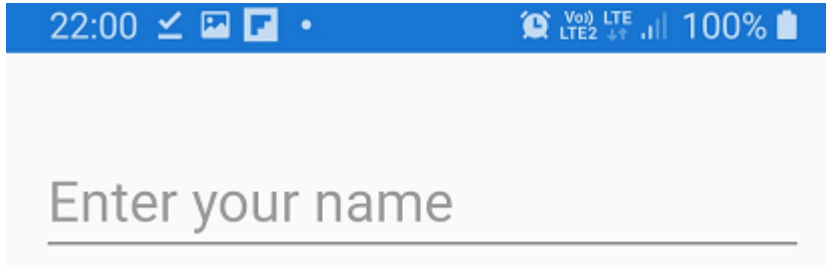
        Odwołania: 0
        protected override void OnSleep()
        {
            Console.WriteLine("OnSleep");
        }

        Odwołania: 0
        protected override void OnResume()
        {
            Console.WriteLine("OnResume");
        }
    }
}
```
- Annotations:** Three orange text annotations with arrows pointing to specific code blocks:
 - "aplikacja uruchamia się" points to the `OnStart()` method.
 - "aplikacja przechodzi w tryb pracy w tle" points to the `OnSleep()` method.
 - "wznowienie działania aplikacji" points to the `OnResume()` method.
- Output Window:** Shows the following log output:

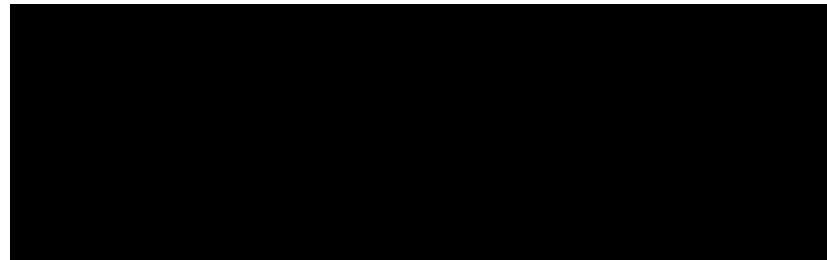
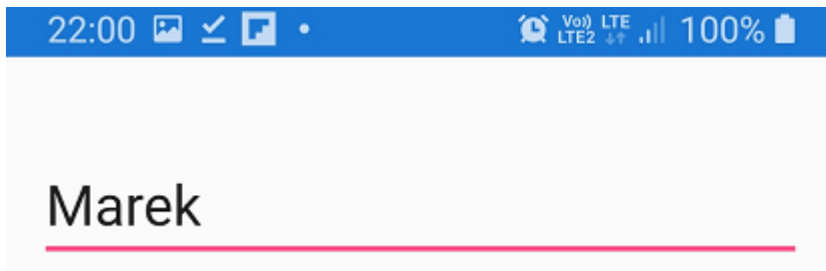
```
Dane wyjściowe
Pokaz dane wyjściowe z: Debugowanie
[ViewRootImpl@de62728[MainActivity]] stopped(true) old=false
OnSleep
[ViewRootImpl@de62728[MainActivity]] MSG_WINDOW_FOCUS_CHANGED 0 1
[InputMethodManager] prepareNavigationBarInfo() DecorView@78bc2e4[MainActivity]
[InputMethodManager] getNavigationBarColor() -855310
```
- UI Elements:** The status bar at the bottom shows "Nie znaleziono żadnych problemów" and "Wartość". The bottom right corner has a toolbar with "Dodaj do kontroli źródła" and "Wybierz repozytorium".

Samouczek cyklu życia aplikacji

Cykl życia aplikacji

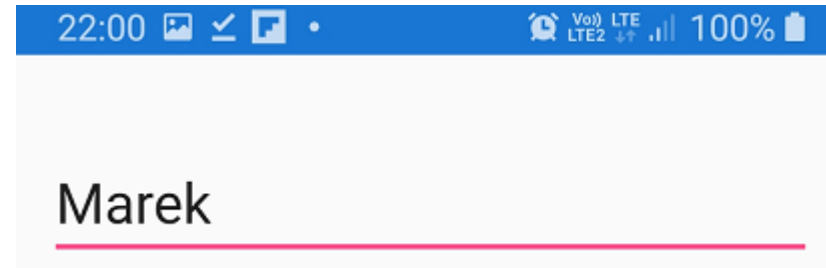


aplikacja uruchamia się



aplikacja przechodzi w tryb pracy w tle

zapamiętujemy pole Name w
tablicy Properties pod
indeksem keyName



wznowienie działania aplikacji

```
using System;
using Xamarin.Forms;
using Xamarin.Forms.Xaml;
```

App.xaml.cs

Cykl życia aplikacji

namespace ApplicationLifecycle

```
{
    Odwołania: 7
    public partial class App : Application
    {
        // pole klasy
        Odwołania: 4
        public string Name { get; set; }

        // klucz
        const string keyName = "keyName";

        1 odwołanie
        public App()
        {
            InitializeComponent();

            MainPage = new MainPage();
        }

        Odwołania: 0
        protected override void OnStart()
        {
            Console.WriteLine("OnStart");

            // przywrócenie pola Name (jeśli istnieje klucz keyName)
            if (Properties.ContainsKey(keyName))
            {
                Name = (string)Properties[keyName];
            }
        }

        Odwołania: 0
        protected override void OnSleep()
        {
            Console.WriteLine("OnSleep");
            // zapamiętujemy pole Name w tablicy Properties pod indeksem keyName
            Properties[keyName] = Name;
        }

        Odwołania: 0
        protected override void OnResume()
        {
            Console.WriteLine("OnResume");
        }
    }
}
```

Nie jest konieczne przywrócenie danych ze słownika Properties w przeciążeniu metody OnResume. Jest to spowodowane tym, że gdy aplikacja zaczyna działać w tle, ona i jej stan są nadal w pamięci.

MainPage.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="ApplicationLifecycle.MainPage">
    <StackLayout Margin="20,35,20,20">
        <Entry x:Name="entry"
              Placeholder="Enter your name"
              Completed="OnEntryCompleted" />
    </StackLayout>
</ContentPage>
```

MainPage.xaml.cs

```
using System;
using Xamarin.Forms;

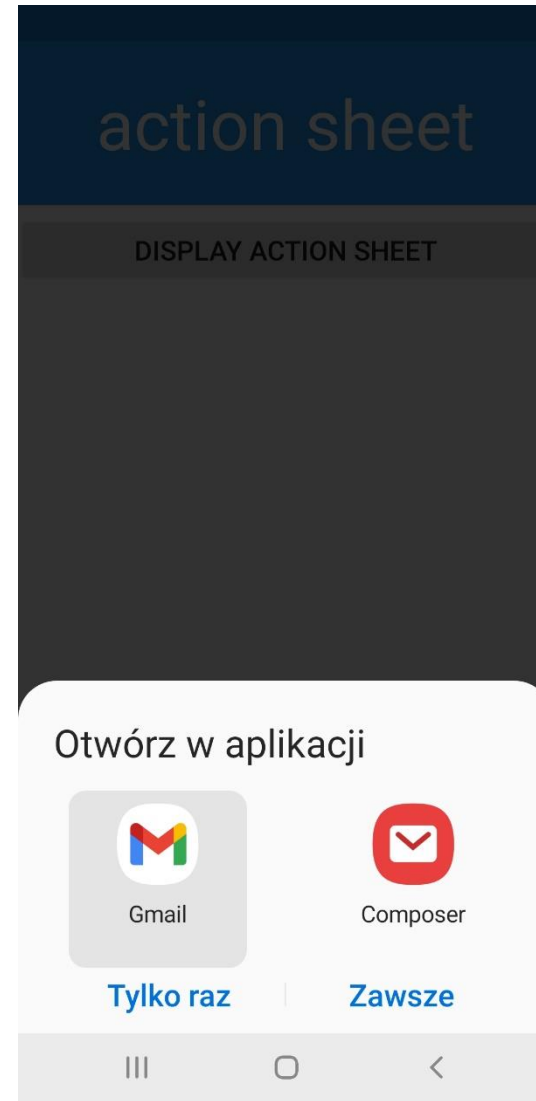
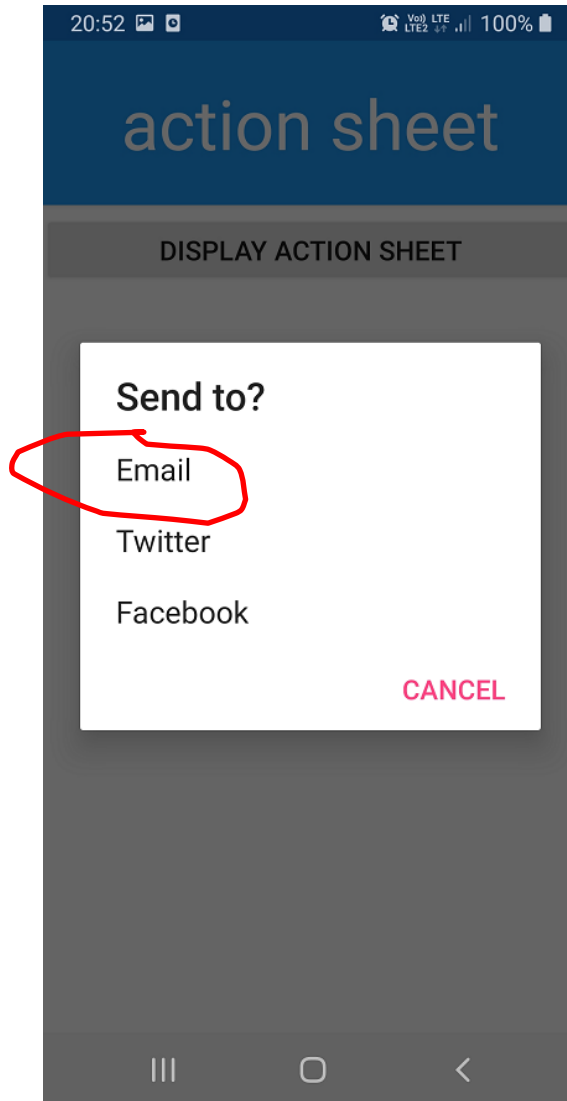
namespace ApplicationLifecycle
{
    Odwołania: 5
    public partial class MainPage : ContentPage
    {
        1 odwołanie
        public MainPage()
        {
            InitializeComponent();
        }

        // metoda wywołwana tuż przed ukazaniem okna aplikacji
        Odwołania: 0
        protected override void OnAppearing()
        {
            base.OnAppearing();

            // zawartość pola Name jest wstawiana do kontrolki entry
            entry.Text = (Application.Current as App).Name;
        }

        // po wpisaniu imienia jest ono zapamiętane w polu Name
        Odwołania: 0
        void OnEntryCompleted(object sender, EventArgs e)
        {
            (Application.Current as App).Name = entry.Text;
        }
    }
}
```

Arkusz akcji



po wybraniu akcji
odpowiednia
aplikacja
uruchamia się
(jeśli jest
zainstalowana)

[Wyświetlanie arkusza akcji](#)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;
using Xamarin.Essentials; // Launcher
```

MainPage.xaml.cs

```
namespace actionSheet
```

```
{
    Odwołania: 5
    public partial class MainPage : ContentPage
    {
        1 odwołanie
        public MainPage()
        {
            InitializeComponent();
        }

        Odwołania: 0
        async void OnDisplayActionSheetButtonClicked(object sender, EventArgs e)
        {
            // wyświetlamy action sheet
            // metoda DisplayActionSheet jest asynchroniczna, czekamy na wynik jej działania
            string action = await DisplayActionSheet("Send to?", "Cancel", null, "Email", "Twitter", "Facebook");

            // uruchamiamy Facebooka (jeśli jest zainstalowany)
            if (action == "Facebook")
            {
                await Launcher.OpenAsync(new Uri("fb://page/page_id"));
                /*
                // gdy Facebook jest niezainstalowany uruchamiamy przeglądarkę
                if (!await Launcher.TryOpenAsync("fb://page/page_id"))
                {
                    await Browser.OpenAsync("https://www.facebook.com/");
                }
                */
            }

            // uruchamiamy Twittera (jeśli jest zainstalowany)
            else if (action == "Twitter")
            {
                await Launcher.OpenAsync(new Uri("twitter://user?user_id=userid"));
                // gdy Twitter jest niezainstalowany uruchamiamy przeglądarkę
                //if (!await Launcher.TryOpenAsync("twitter://user?user_id=userid"))
                //{
                //    await Browser.OpenAsync("https://twitter.com/");
                //}
            }

            // uruchamiamy klienta pocztowego (jeśli jest zainstalowany)
            else if (action == "Email")
            {
                var address = "adres@onet.pl";
                var subject = "Tytuł maila";
                var content = "Treść maila";
                Uri uri = new Uri($"mailto:{address}?subject={subject}&body={content}");
                await Launcher.OpenAsync(uri);
            }
        }
    }
}
```

Arkusze akcji

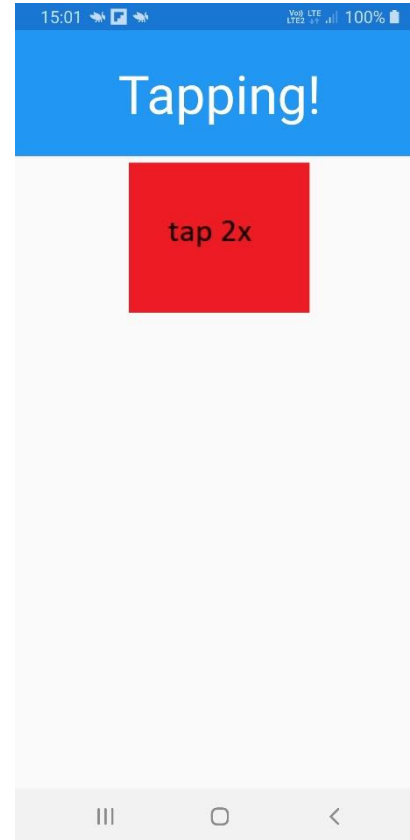
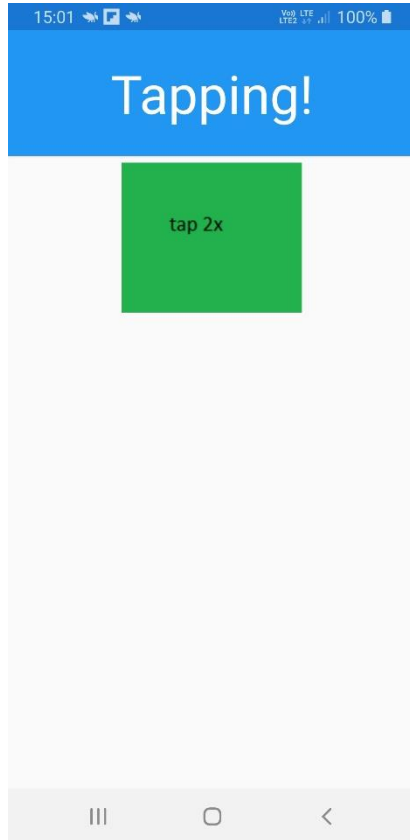
MainPage.xaml

```
<?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="actionSheet.MainPage">

    <StackLayout>
        <Frame BackgroundColor="#2196F3" Padding="24" CornerRadius="0">
            <Label Text="action sheet" HorizontalTextAlignment="Center" TextColor="White" FontSize="36"/>
        </Frame>
        <Button Text="Display action sheet"
              Clicked="OnDisplayActionSheetButtonClicked" />
    </StackLayout>
</ContentPage>
```

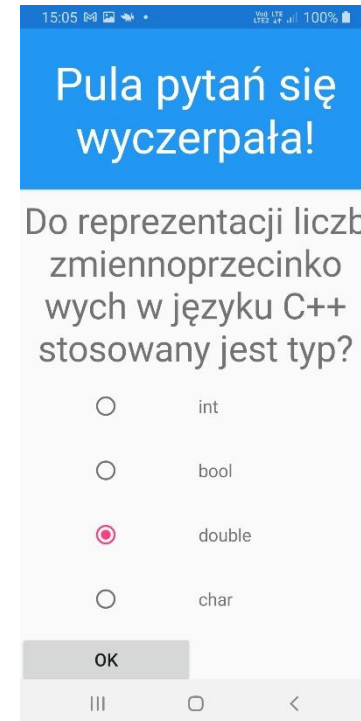
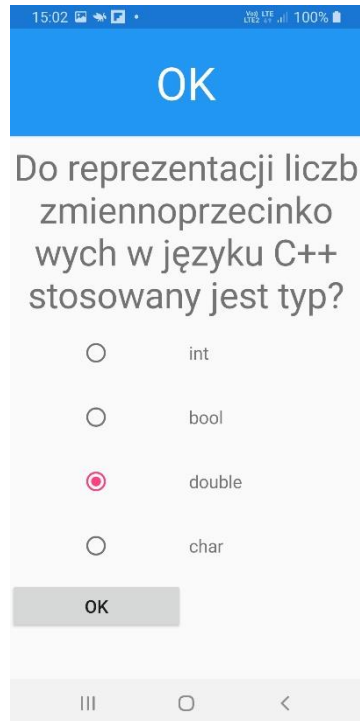
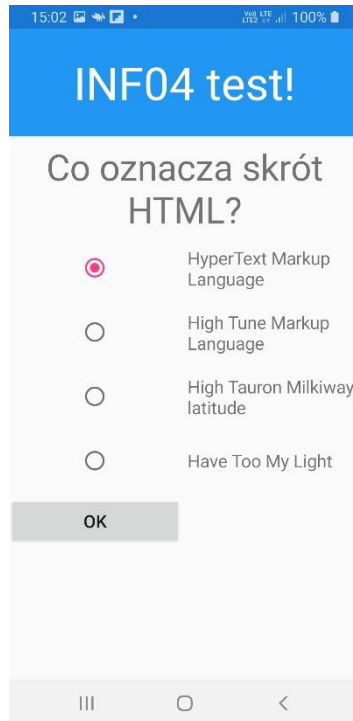
Uniform Resource Identifier
(Ujednolicony Identyfikator Zasobów)

Tapping



<https://learn.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/gestures/tap>

INF04 test



cukrzyca (wyliczanie dawek insuliny)

```
#include <iostream>
using namespace std;

// deklaracja funkcji
double myGlukoza(double, double, int);

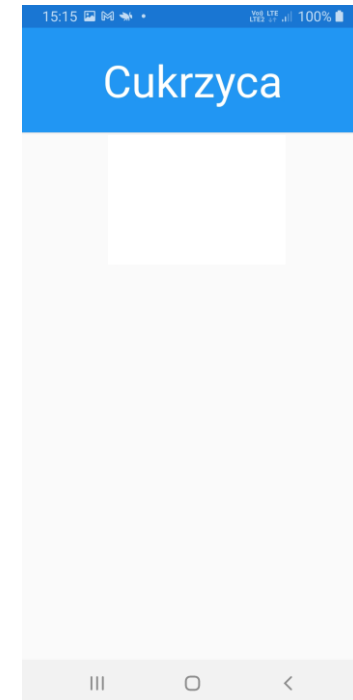
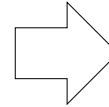
int main()
{
    cout << "Kalkulator cukrzycowy\n\n";

    // deklaracja zmiennych
    double glukoza = 0;
    double dziesiatki = 0;
    double insulina = 0;
    int posilek = 0;
    int jednostki = 0;

    cout << "podaj ilosc glukozy:\n";
    cin >> glukoza;
    cout << "podaj ilosc posilkow (ilosc dziesiatek)\n";
    cin >> dziesiatki;

    cout << "wybierz posilek:\n\n";

    cout << "[1] sniadanie (7 jednostek)" << endl;
    cout << "[2] drugie sniadanie (14 jednostek)" << endl;
    cout << "[3] obiad (14 jednostek)" << endl;
    cout << "[4] podwieczorek (6 jednostek)" << endl;
    cout << "[5] kolacja (10 jednostek)" << endl;
    cout << "[6] druga kolacja (10 jednostek)" << endl;
}
```



<https://docs.microsoft.com/pl-pl/xamarin/xamarin-forms/xaml/xaml-controls>

kontrolki

<https://github.com/xamarin/xamarin-forms-samples>

github przykłady

<https://learn.microsoft.com/pl-pl/xamarin/get-started/tutorials/>

samouczki