NMCT

Domotica

Xamarin Cross Platform Mobile App Dev.

Xamarin

Xamarin

- Platformen
  - Apple (Swift/Objective C)
    - iOS
      - iPhone
      - iPad
    - Watch OS
      - Apple Watch
    - tvOS
      - Apple Tv
  - Google (Java)
    - Android
      - Smartphones
    - Android Wear
      - Smartwatches

Xamarin

- Microsoft (C# UWP)
  - Windows 10
    - Desktop
    - Tablet
    - Phone
    - Xbox
    - IoT Core

Xamarin

- Conclusie
  - 3 platformen
    - 3 x ontwikkelen
    - 3 x testen
    - 3 x bugfixen
    - 3 x updaten
    - 3 x verschillende competenties bij werknemers (Swift/Java/.NET)
    - 3 x zo duur

Xamarin

- Oplossing
  - Xamarin (.NET C#)
  - Cordova (HTML/CSS/JS)
  - Appcelerator Titanium (HTML/CSS/JS)
  - ..

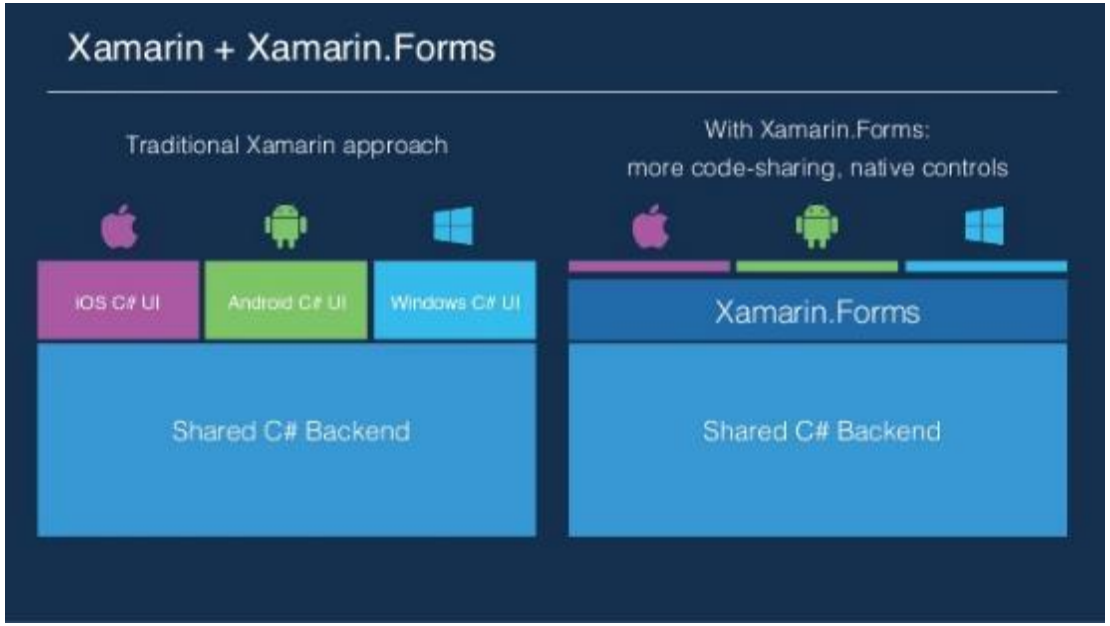
- 1 code base => meerdere platformen is doelstelling
  - Niet altijd zo
  - Veel valkuilen
  - Blijft lasting verhaal

Xamarin

- Wij bekijken Xamarin
  - U mag ook iets anders kiezen

Xamarin

- Xamarin
  - 3 platformen
    - Xamarin.iOS
    - Xamarin.Android
    - Xamarin Windows
  - 3 x UI ontwikkelen
  - 1 x backend
  - Alles in C# of F#
  - Je moet de platform eigenschappen kennen (ervaring met Swift/Java/C# is zeer handig)
  - Meeste native ervaring

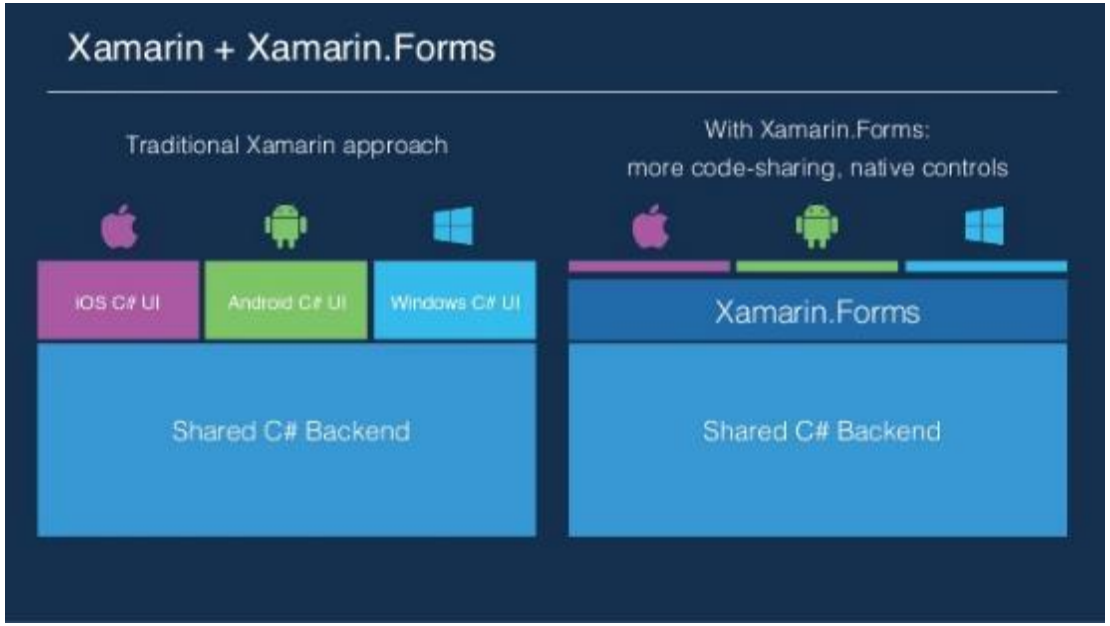


Xamarin

Xamarin.Forms

- 1 x UI ontwikkelen
  - XAML
  - Xamarin Forms zal look en feel toepassen van OS
- 1 x backend
- Enkel OS specifieke zaken kan je niet delen
  - Push/Camera/…

- Wij kiezen voor Xamarin.Forms



Xamarin

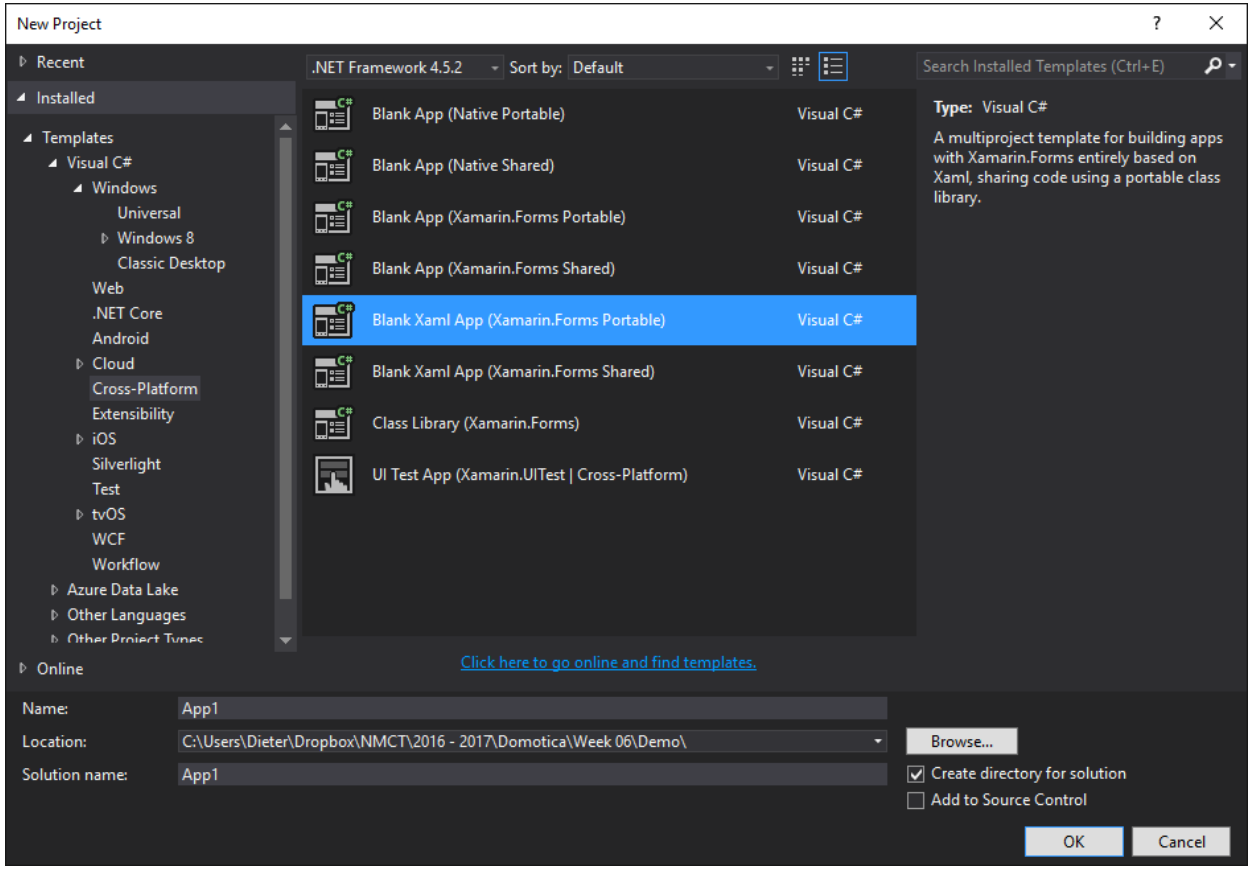
- Wat heb je nodig ?
  - iOS
    - Mac of PC (Deployment enkel via Mac mogelijk, iOS Simulator wel mogelijk op Windows mits een mac in network)
    - Op de Mac moet Xcode staan
    - Xamarin Studio / Visual Studio 2015
  - Android
    - Mac of Pc
    - Android SDK
    - Xamarin Studio/Visual Studio 2015
  - Windows
    - PC
    - Windows 10 SDK
    - Xamarin Studio/Visual Studio 2015

Xamarin

- Testen
  - iOS
    - Microsoft iOS Emulator (Windows , maar Mac in network)
    - Apple iOS (OSX)
    - Toestel (hiervoor Appe Dev. Account nodig $$)
  - Android
    - Microsoft Android Emulator (Windows)
    - Google Android Emulator (Windows en OSX)
    - Toestel (geniet voorkeur)
  - Windows
    - Microsoft Windows Phone Simulator (Windows)
    - Toestel (geniet voorkeur)

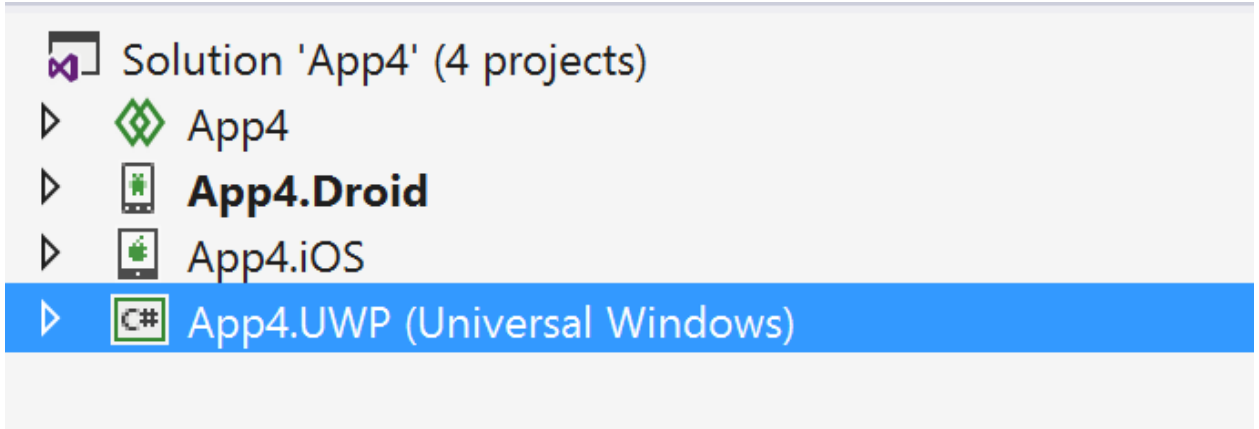
Xamarin

- Nieuw project



Xamarin

- Hangt af van de geïnstalleerde SDK's welke projecte je ziet

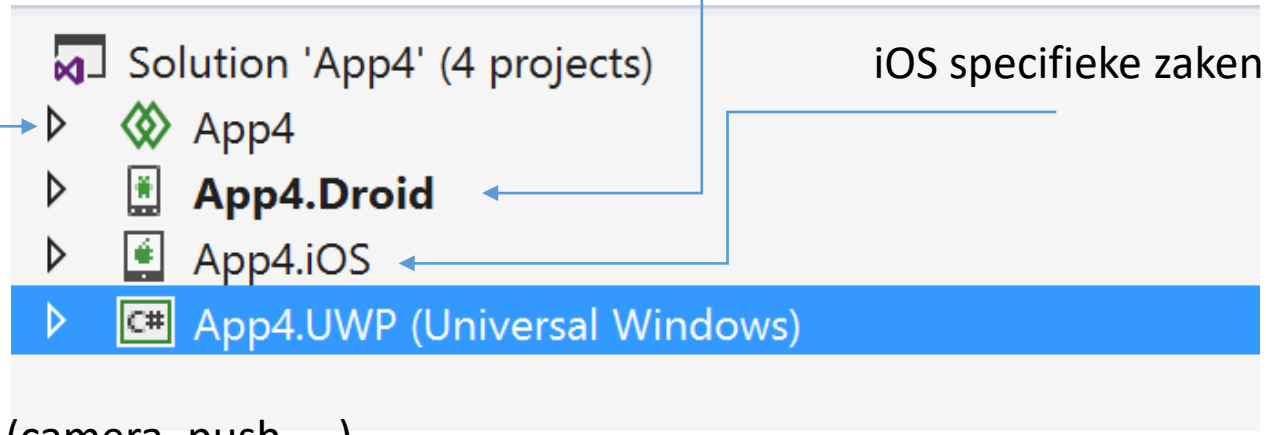


Xamarin

Gedeelde code met alle platformen, ook UI (XAML)

Android specifieke zaken (camera, push, …)

iOS specifieke zaken (camera, push, …)



Windows specifieke zaken (camera, push, …)

Xamarin

- Leercuve
  - Wat moet je kennen ?
    - C# ✓
    - .NET ✓
    - XAML ✓
    - Platform specifieke zaken
      - Camera aanspreken
      - Push
      - HTTP Client ✓
      - Bluetooth
      - …
    - Documentatie
      - Quickstarts
      - Veel code examples
      - Mooie community
    - Learn by doing

Xamarin

- Quickstart 1 (Samen)
  - https://developer.xamarin.com/guides/xamarin-forms/getting-started/hello-xamarin-forms/quickstart/

Xamarin

- Basis is dependency injection
  - Interface definiëren (IDial in dit voorbeeld) in het “shared project”
  - Per platform implementatie van IDial

Registratie in Dependency Container

```
[assembly: Dependency(typeof(PhoneDialer))]
namespace Phoneword.Droid {
    1 reference
    public class PhoneDialer : IDialer {
        0 references
        public bool Dial(string number) {
            var context = Forms.Context;
            if (context == null)
                return false;

            var intent = new Intent(Intent.ActionCall);
            intent.SetData(Uri.Parse("tel:" + number));

            if (IsIntentAvailable(context, intent)) {
                context.StartActivity(intent);
                return true;
            }

            return false;
        }
```

```
[assembly: Dependency(typeof(PhoneDialer))]
namespace Phoneword.iOS {
    1 reference
    public class PhoneDialer : IDialer {
        0 references
        public void Demo(string fromOs) {
            UIAlertView _error = new UIAlertView("bericht",fromOs, null, "Ok", null);
            _error.Show();
        }
```

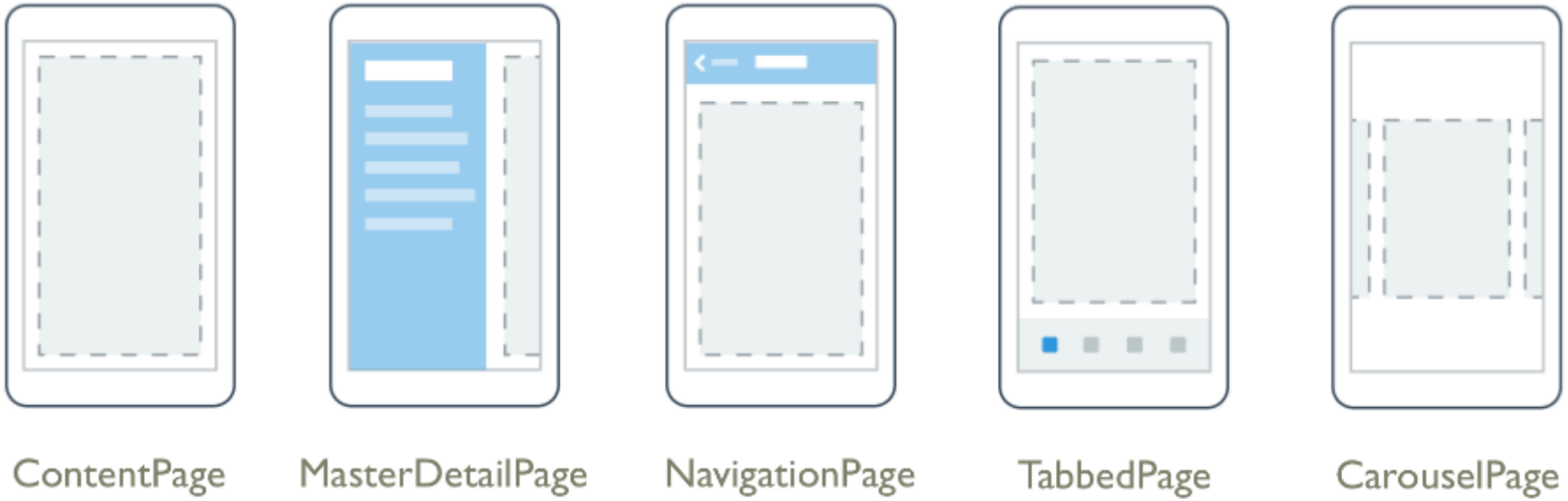
Xamarin

- Aanroepen methode in shared project
  - DependencyService.Get<… Inteface …>

```
0 references
async void OnCall(object sender, EventArgs e) {
    if (await this.DisplayAlert(
            "Dial a Number",
            "Would you like to call " + translatedNumber + "?",
            "Yes",
            "No")) {
        var dialer = DependencyService.Get<IDialer>();
        if (dialer != null)
            dialer.Dial(translatedNumber);
    }
}
```

Xamarin

- https://developer.xamarin.com/guides/xamarin-forms/user-interface/navigation/



Xamarin

- Controls
  - The Usual Suspects
    - Button
    - Textbox
    - Listview
    - …
  - Extra controls
    - Maps
    - Webview
    - TableView
    - …

Xamarin

- XAML
  - Kennen we van WPF & Windows UWP
  - Databinding aanwezig, zelfde manier van werken
  - Styling aanwezig, ongeveer zelfde manier van werken

Xamarin

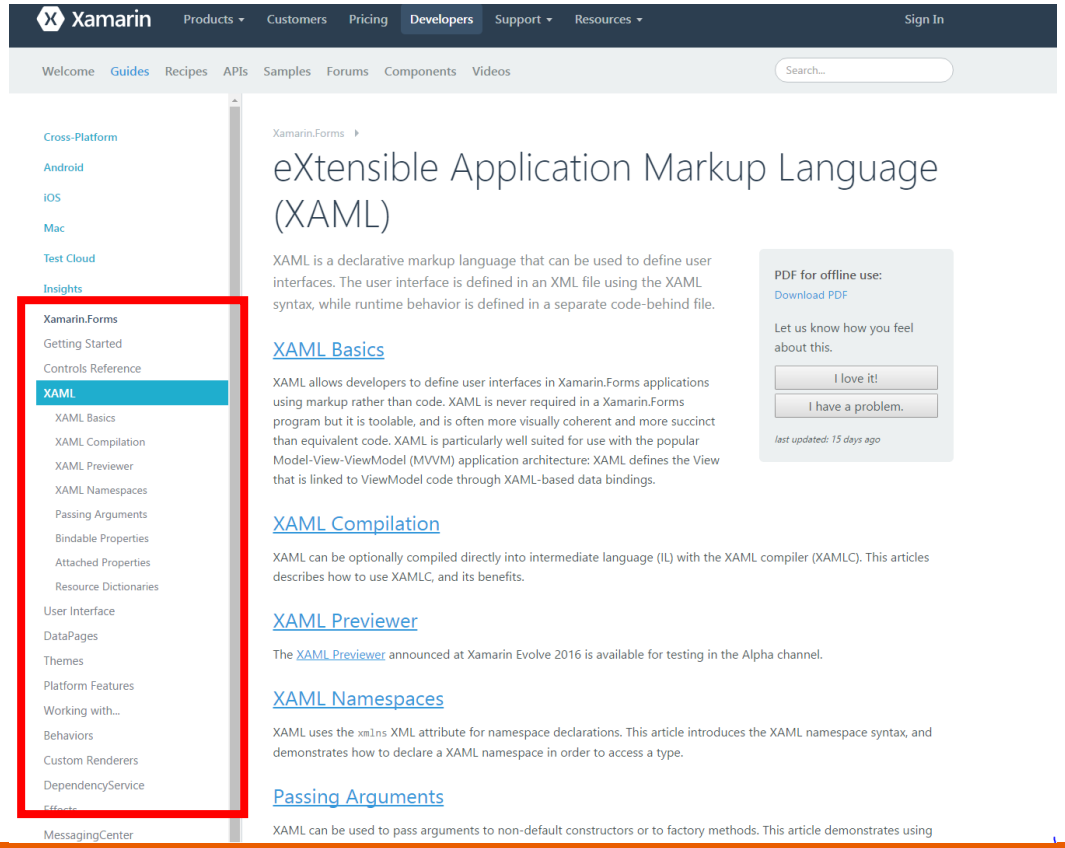
- Webservices
  - Gebruik HTTP Client
  - Gebruik JSON.NET

Xamarin

- MVVM
  - Aan te raden manier voor apps die lang onderhoudbaar moeten zijn
  - Valt buiten de scope van dit vak
  - Indien iteresse
    - https://developer.xamarin.com/guides/xamarin-forms/xaml/xaml-basics/data_bindings_to_mvvm/
    - MVVM Light
    - MVVM Cross
  - Geen verplichting voor project !!

Xamarin

- https://developer.xamarin.com/guides/xamarin-forms/xaml/



Xamarin

- Opdracht
  - Maak een Tabed Navigation Page Application
  - Op Tab 1: opvragen van alle temperatuur registraties
  - Op Tab 2: temperatuur doorsturen
  - We maken gebruik van volgende webservice uit de vorige lessen:
    - http://domotica-nmct.azurewebsites.net/api/pi
    - Zie klasse TemperatuurData voor ontvangen en versturen van data
    - PiName is uw eigen unieke naam

```
5 references
public class TemperatuurData {
    1 reference
    public decimal Value { get; set; }
    1 reference
    public string Sensor { get; set; }
    1 reference
    public string PiName { get; set; }
}
```

Xamarin

- Wat met cross platform hardware features ?
  - Foto/video
  - Push
  - Bluetooth
  - …
- Je kan zelf implementeren in platform project en via DI aanspreken in het shared project
- Meestal wel een plugin beschikbaar
  - Vb xam.plugins….
  - Bekijk zelf een https://github.com/jamesmontemagno/MediaPlugin
  - Probeer een foto te nemen en deze weer te geven in de applicatie
  - Plugins zoveel mogelijk toevoegen via
    - Nuget
    - Components

Xamarin

- Tips
  - Herstarten Visual Studio kan veel helpen
  - Clean & Build
  - Bij Android
    - Check goed welke API je target (versie van Android)
  - Als je functionaliteit nodig hebt zoek ALTIJD
    - Is er een nuget package?
    - Is er een component?
  - Voor je project check zeker xam.plugins….