



Desarrollo de Apps con Xamarin Forms.

Curso básico de introducción.

Por Angel Garcia.



Programa del curso.

Módulo 1.

- Qué es Xamarin Forms.
- Requerimientos de Hardware y Software.
- Creando nuestra primera app..
- Views & Controls.

Módulo 2.

- Interfaces de usuario en Xamarin Forms (Layouts & Pages).
 - XAML.
 - Code behind.
- Navigation.
- Data Binding.
- Patrón Model-View-ViewModel (MVVM).

Módulo 3.

- ListViews.
- Behaviours.
- Triggers.
- Effects.
- Renderers.

Módulo 4.

- Inyección de dependencias.
- Introducción a los frameworks MVVM en Xamarin Forms.
- Introducción a Prism.
- Consumiendo data desde un API Rest.

Módulo 5.

- Dependency Services.
- Usando funcionalidades nativas con plugins.
- Persistencia local con Realm.

Módulo 6.

- Configuraciones específicas de iOS & Android.
- Visual Studio App Center (Crashes, Analytics, Building, Distribution).



Semana 2.

Interfaces de usuario en Xamarin Forms (XAML).

Pages & Layouts.

Navigation.

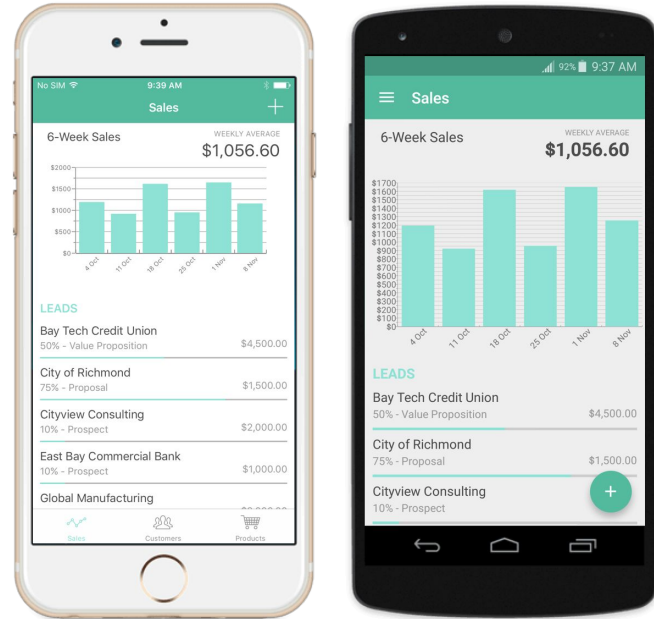
Data Binding.

Patrón Model-View-ViewModel (MVVM).

Interfaces de usuario

El UI en Xamarin Forms se puede definir de dos formas, por XAML o por code behind (C#).

Usaremos XAML para separar la capa de presentación de la lógica en nuestras aplicaciones.





XAML.

XAML (Extensible Application Markup Language) es un lenguaje de marcado basado en XML que se usa para la creación de interfaces de usuario en Xamarin.Forms, sin necesidad de utilizar código.

XAML

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  x:Class="XamlSamples.HelloXamlPage"
  Title="Hello XAML Page">
  <ContentPage.Content>

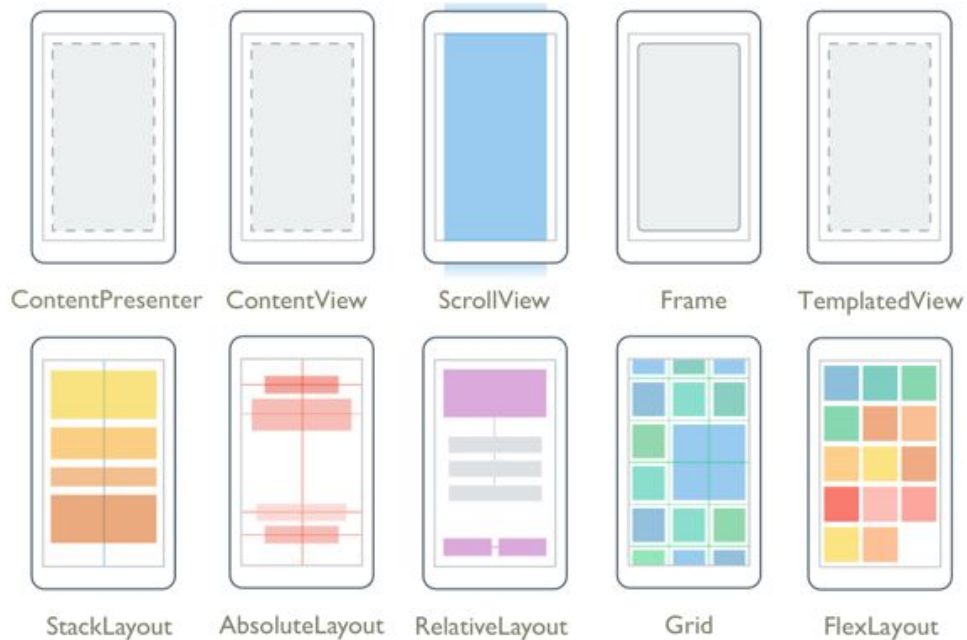
    <Label Text="Hello, XAML!"
      VerticalOptions="Center"
      HorizontalTextAlignment="Center"
      Rotation="-15"
      IsVisible="true"
      FontSize="Large"
      FontAttributes="Bold"
      TextColor="Blue" />

  </ContentPage.Content>
</ContentPage>
```

Layouts.

Los layouts son elementos **contenedores** de otros layouts o vistas (controles), son especialmente necesarios ya que las páginas sólo pueden contener un elemento hijo.

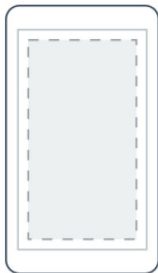
Los utilizaremos para establecer la posición y alineación de los elementos que contienen.



Pages.

Las páginas son elementos contenedores que representan una pantalla de la aplicación.

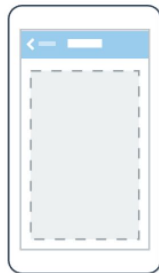
Xamarin.Forms.Page representa un **ViewController** en iOS, un **Page** en UWP, en Android cada página se comporta como un **Activity**, pero no lo son.



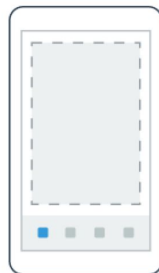
ContentPage



MasterDetailPage



NavigationPage



TabbedPage



TemplatedPage



CarouselPage

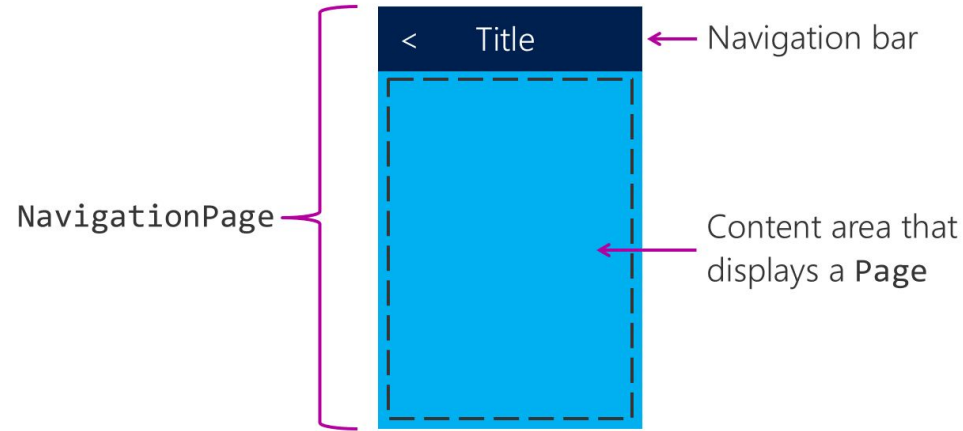


Navigation.

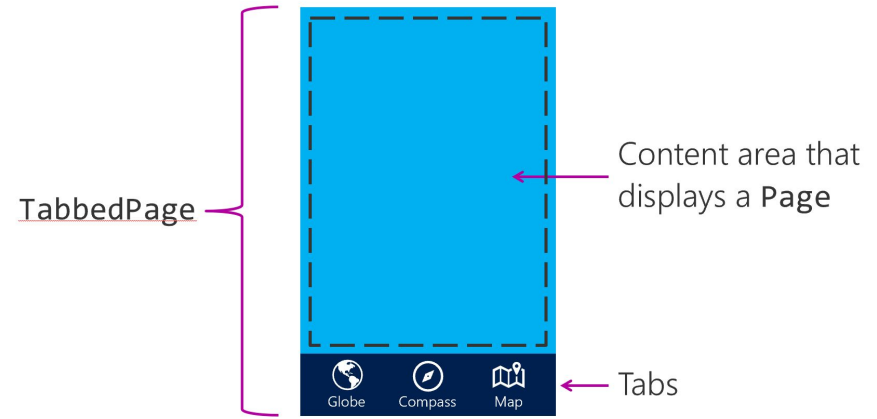
La interfaz de usuario de la mayoría de las aplicaciones se compone de varias páginas. Cuando usamos el término navegación, nos referimos al conjunto de transiciones entre estas páginas.

La navegación incluye tanto la UI como las acciones del usuario necesarias para llevar al usuario de una página a otra.

Stack Navigation



Tab Navigation



Data Binding

El **data binding** conecta dos objetos, llamados **source**(origen) y **target**(destino). El objeto origen proporciona los datos, mientras que el objeto de destino, que debe ser una propiedad enlazable (**bindable property**), consumirá (y a menudo visualizará) datos del objeto de origen. Por ejemplo, un **Label** (objeto de destino) generalmente vinculará su propiedad **Text** a una propiedad string pública en un objeto fuente.





- 1 Raise PropertyChanged notification



```
t.Notes = "Get Groceries";
```

Binding

- 3 Binding updates target property

Name Pickup some Milk

Notes Get Groceries

Done



- 2 Binding reads new property value

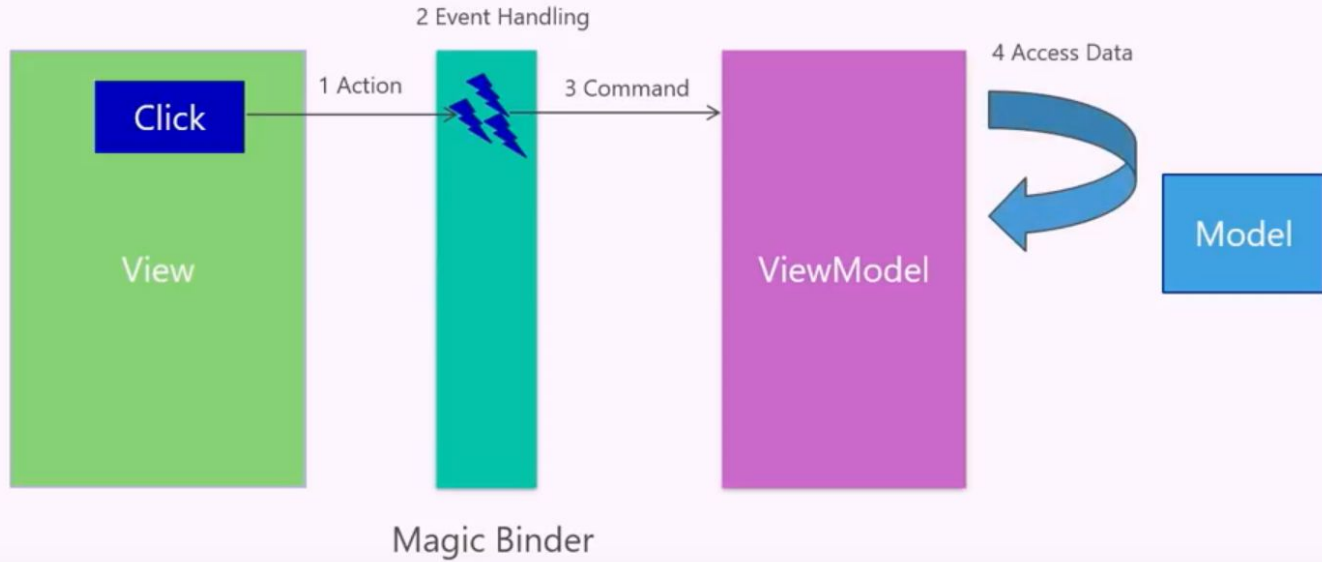
M-V-VM.

Se trata de un patrón de desarrollo que nos permite separar o desacoplar la interfaz de usuario del resto del código. Podemos dividir nuestra aplicación en tres capas:



MVVM **MODEL VIEW VIEWMODEL**

Data + Commanding





Lectura pa' la casa.

Ejemplos del repositorio de Xamarin Forms

<https://github.com/davidbritch/xamarin-forms/tree/master/DataBinding101>

<https://github.com/davidbritch/xamarin-forms/tree/master/Commanding>

Ejemplos XAML

<https://github.com/xamarin/xamarin-forms-samples/tree/master/FormsGallery/FormsGallery/FormsGallery/XamlExamples>

Documentación oficial de las Pages

<https://docs.microsoft.com/en-us/xamarin/xamarin-forms/user-interface/controls/pages>

Documentación oficial de los Layouts

<https://docs.microsoft.com/en-us/xamarin/xamarin-forms/user-interface/controls/layouts>

Introducción a Data Binding

<https://blog.xamarin.com/introduction-to-data-binding/>

MVVM en Xamarin Forms

https://www.youtube.com/watch?v=vQBrMoh_HSQ

<https://ramonesteban78.github.io/es/tutorials/mvvm/xamarin-forms-and-mvvm.html>



Microsoft Learn Recommendations

Xamarin Forms 101

<https://docs.microsoft.com/en-us/learn/modules/create-a-mobile-app-with-xamarin-forms/>

XAML

<https://docs.microsoft.com/en-us/learn/modules/create-ui-in-xamarin-forms-apps-with-xaml/>

Navigation

<https://docs.microsoft.com/en-us/learn/modules/create-multi-page-xamarin-forms-apps-with-stack-and-tab-navigation/index>

MVVM

<https://docs.microsoft.com/en-us/learn/modules/design-a-mvvm-viewmodel-for-xamarin-forms/>

Data Binding

<https://docs.microsoft.com/en-us/learn/modules/separate-your-ui-and-code-with-xamarin-forms-data-binding/index>

Layouts and Pages

<https://docs.microsoft.com/en-us/learn/modules/customize-layout-in-xamarin-forms-xaml-pages/>



Gracias por su atención.



Sus preguntas, dudas o sugerencias son bienvenidas.

angelrenegarcia13@gmail.com

<https://github.com/AngelGarcia13/XamarinForms101Class2019>