XAMARIN

**SEC-1: GETTING STARTED**

**CREATING A XAMARIN FORMS APP**

Creamos una aplicación mobile de xamarin en “Crear proyecto”. Luego cuando tengamos el proyecto creado hacemos click derecho en la carpeta del nombre que le dimos al proyecto y elegimos **Administrar paquetes NuGet**. Actualizamos todos los paquetes.

Luego elegimos la carpeta con el tipo de app que queremos (Android, IOS, etc.), y presionamos **ctrl + F5** para descargar el emulador.

**ACCESING ELEMENTS**

**GreetPage.xaml.cs:**

public partial class GreetPage : ContentPage

{

public GreetPage()

{

InitializeComponent();

slider.Value = 0.5;

}

private void Slider\_ValueChanged(object sender, Xamarin.Forms.ValueChangedEventArgs e)

{

label.Text = String.Format("Value is {0:F2}", e.NewValue);

}

}

**GreetPage.xaml:**

<ContentPage

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

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

x:Class="FirstApplication.GreetPage"

>

<StackLayout HorizontalOptions="Center" VerticalOptions="Center">

<Label Text="Hello World" x:Name="label"/>

<Slider ValueChanged="Slider\_ValueChanged" x:Name="slider"/>

</StackLayout>

</ContentPage>

**DATA BINDING**

Utilizamos los **XAML markup extensions**:

<Label Text="{Binding

Source={x:Reference slider},

Path=Value,

StringFormat='Value is {0:F2}'}"

x:Name="label"

/>

**BINDING CONTEXT**

Resumimos el **Source y el Path**.

<ContentPage

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

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

x:Class="FirstApplication.GreetPage"

>

<StackLayout

BindingContext="{x:Reference slider}" HorizontalOptions="Center" VerticalOptions="Center"

>

<BoxView

Color="Green"

Opacity="{Binding Value}"/>

<Label

Text="{Binding Value,

StringFormat='Value is {0:F2}' }"

Opacity="{Binding Value}"

/>

<Slider x:Name="slider"/>

</StackLayout>

</ContentPage>

**DEALING with DEVICE DIFFERENCES**

public GreetPage()

{

InitializeComponent();

slider.Value = 0.5;

//if(Device.OS == TargetPlatform.iOS)

//{

// Padding = new Thickness(0, 20, 0, 0);

//}

Padding = Device.OnPlatform(

iOS: new Thickness(0, 20, 0, 0),

Android: new Thickness(10, 20, 0, 0),

WinPhone: new Thickness(30, 20, 0, 0)

);

}

**PROPERTY ELEMENT SYNTAX**

Utilizamos esta sintaxis cuando estamos lidiando con objetos complejos que no pueden ser representados usando simples strings.