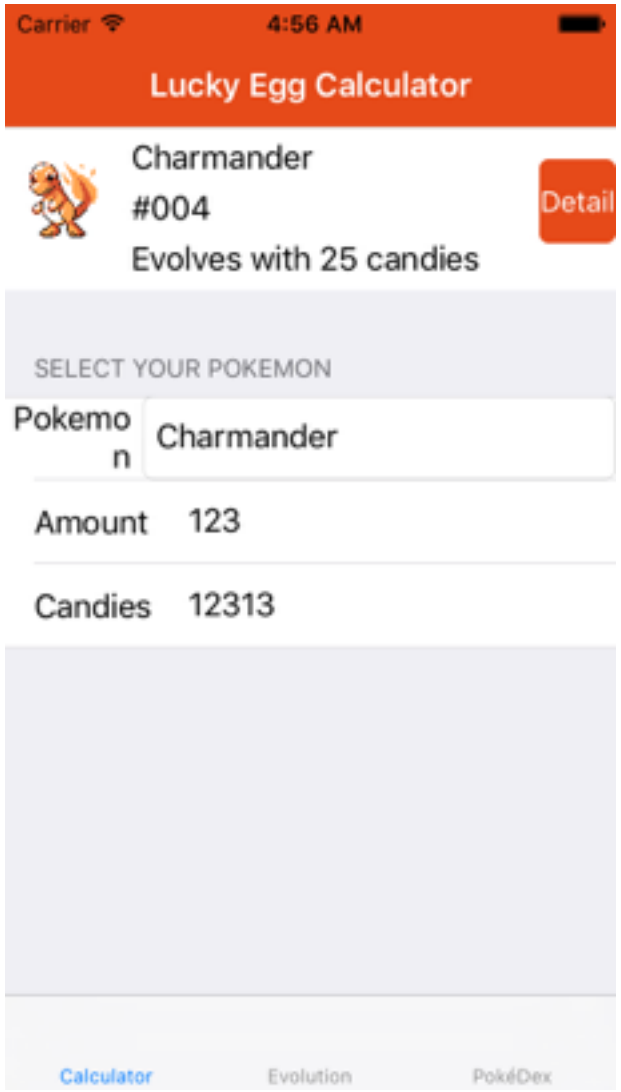

Dev Days

Data Binding y MVVM con Xamarin.Forms

Gustavo de Jesús Barrientos Guerrero
@tavobarrientos
Xamarin Developer

¿Que es MVVM?



- ✓ **Modelo**: Clase que representa los datos.
- ✓ **Vistas**: Elementos de UI.
- ✓ **Vista-Modelo**: Proporciona el estado, acciones y operaciones de la aplicación.

MVVM

Model-View-View Model (MVVM)

Views

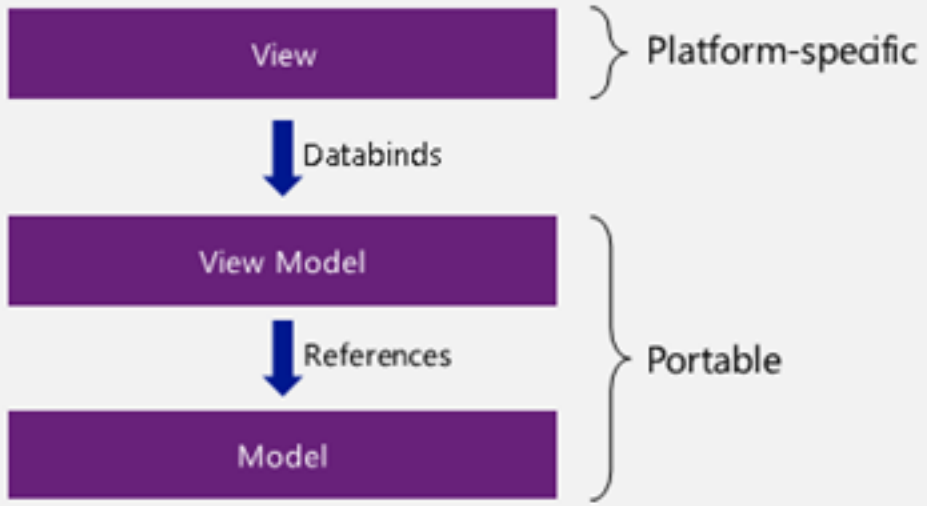
How to display information
Written in XAML

View Models

What information to display
Flow of interaction

Models

Data objects
Business logic
Etc.



Data Binding

Nos sirve para enlazar dos objetos, a menudo se utiliza para mostrar datos.

Data Binding

```xml
<?xml version="1.0" encoding="UTF-8"?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
        xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
        xmlns:local="clr-namespace:PokeCalc;assembly=PokeCalc"
        x:Class="PokeCalc.PokedexPage"
        Title="Pokedex">
    <ContentPage.BindingContext>
        <local:PokedexPageViewModel />
    </ContentPage.BindingContext>

    <ContentPage.Content>
        <ListView
                CachingStrategy="RetainElement"
                ItemTapped="Handle_ItemTapped"
                ItemsSource="{Binding Pokedex}">
            <ListView.ItemTemplate>
                <DataTemplate>
                    <ImageCell Text="{Binding Name}" TextColor="{StaticResource mainColor}" Detail="{Binding Number, StringFormat='#{0}'}"
                </DataTemplate>
            </ListView.ItemTemplate>
        </ListView>
    </ContentPage.Content>
</ContentPage>

```

Para realizar un Databind, utilizamos la propiedad **BindingContext**, que en esencia seria nuestro ViewModel

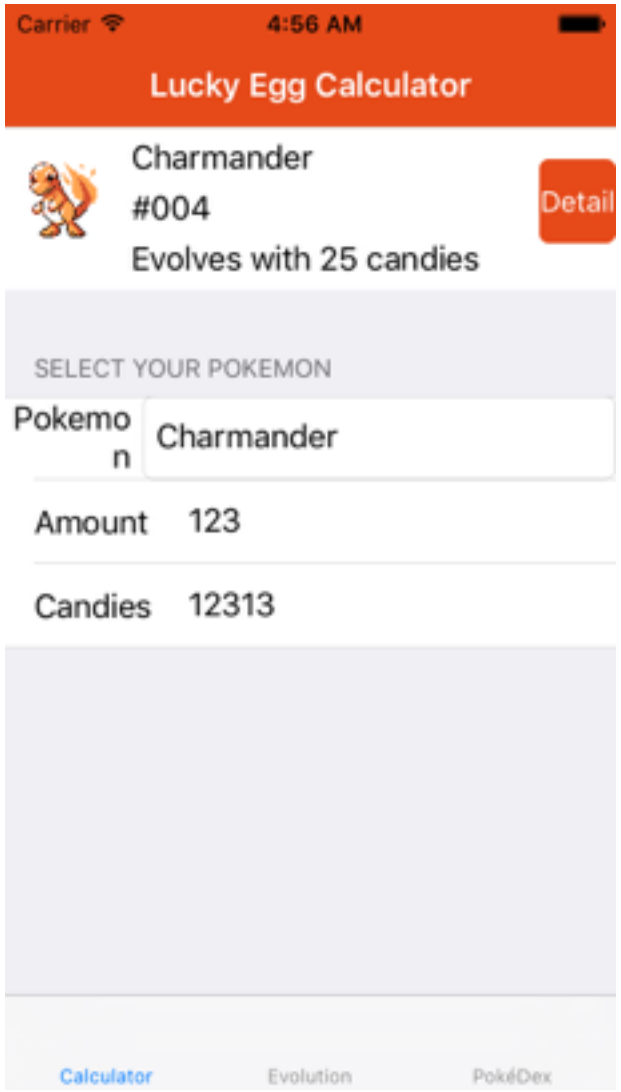
Data Binding

```
using Xamarin.Forms;

namespace PokeCalc
{
    public partial class PokedexPage : ContentPage
    {
        public PokedexPage()
        {
            InitializeComponent();
            BindingContext = new PokedexPageViewModel();
        }
```

Para realizar un Databind, utilizamos la propiedad **BindingContext**, que en esencia seria nuestro ViewModel

Tipos de Data Binding



- ✓ **OneWay**: Los cambios se propagan del origen al destino.
- ✓ **TwoWay**: Los cambios se propagan en ambas direcciones.
- ✓ **OneWayToSource**: Los cambios se propagan del destino al origen.

INotifyPropertyChanged

```
using System.ComponentModel;
using System.Runtime.CompilerServices;

namespace PokeCalc
{
    public class AbstractViewModel : INotifyPropertyChanged
    {
        string _title;
        public string Title
        {
            get
            {
                return _title;
            }
            set
            {
                _title = value;
                OnPropertyChanged();
            }
        }


        public event PropertyChangedEventHandler PropertyChanged;

        protected virtual void OnPropertyChanged([CallerMemberName]string propertyName = null)
        {
            var changed = PropertyChanged;

            if (changed != null)
            {
                PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
            }
        }
    }
}
```

- ✓ Notifica a los elementos que tenemos bindeados a una propiedad, que esta tuvo cambios.
- ✓ Aunque existen paquetes en NuGet que te inyectan está Interfaz a tus propiedades!

DEMO

Código: https://github.com/acidstudios/devdaydemo

Gustavo Barrientos
Guerrero
Xamarin Developer

gustavo.barrientos@acidstudios.me

@tavobarrientos

Gracias

Gustavo Barrientos
Guerrero
Xamarin Developer

gustavo.barrientos@acidstudios.me

@tavobarrientos