



UNIVERSIDAD DE COLIMA

Universidad de Colima

Facultad de Telemática

Ingeniería de software

Programación de Móviles

Ejercicios básicos en Xamarin Forms parte 2

Primera parcial

Almno. Angel Isaac Bejarano Flores

5°D

Mtro. Armando Román Gallardo

Martes 27 de octubre de 2020. Manzanillo, Col.

1. Programa que utiliza el control Picker para seleccionar un elemento de una lista de valores

Codigo 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"
             x:Class="CalculadoraPicker.MainPage">

    <StackLayout>
        <!-- Place new controls here -->
        <Frame BackgroundColor="BlueViolet">
            <Label Text="Calculadora" HorizontalTextAlignment="Center"
                FontSize="Large" TextColor="White"/>
        </Frame>

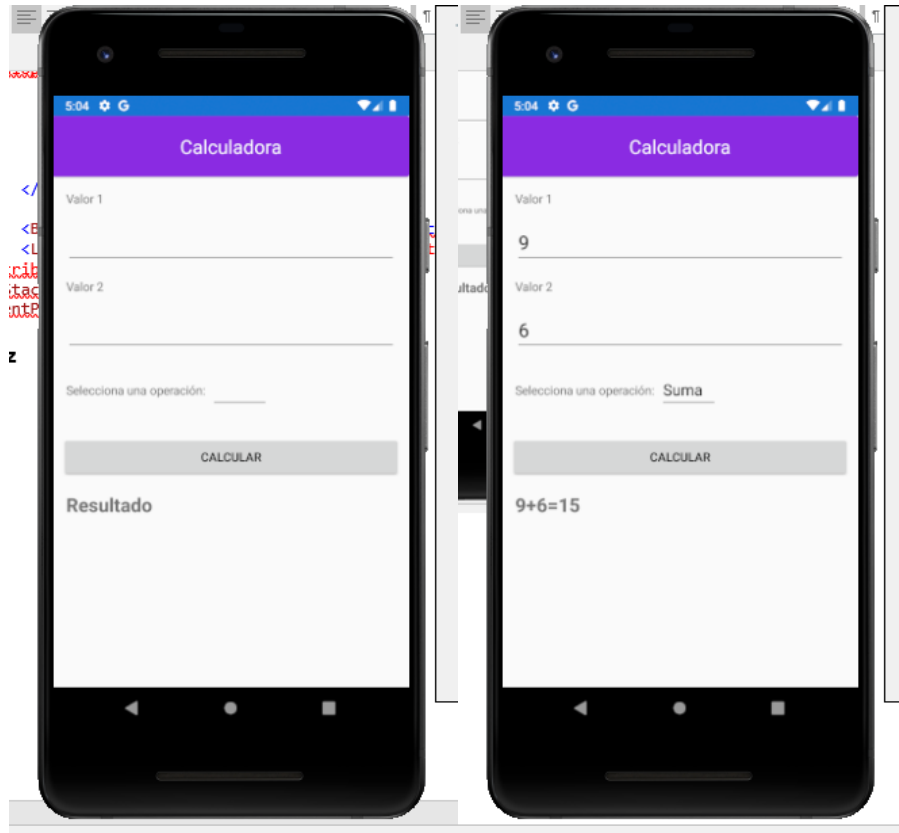
        <Label Text="Valor 1" Margin="15,10"/>
        <Entry x:Name="valor1" Margin="15,0" FontSize="Large" Keyboard="Numeric"/>

        <Label Text="Valor 2" Margin="15,10"/>
        <Entry x:Name="valor2" Margin="15,0" FontSize="Large" Keyboard="Numeric"/>

        <StackLayout Orientation="Horizontal" Margin="15">
            <Label Text="Selecciona una operación:" VerticalOptions="Center"/>
            <Picker x:Name="operacion" Title="
                "
                VerticalOptions="Center">
                <Picker.Items>
                    <x:String>Suma</x:String>
                    <x:String>Resta</x:String>
                </Picker.Items>
            </Picker>
        </StackLayout>

        <Button Text="Calcular" Clicked="Button_Clicked" Margin="10"/>
        <Label x:Name="resultado" Margin="15,0" Text="Resultado"
            FontAttributes="Bold" FontSize="Large"/>
    </StackLayout>
</ContentPage>
```

Interfaz



Codigo C#

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

namespace CalculadoraPicker
{
    public partial class MainPage : ContentPage
    {
        public MainPage()
        {
            InitializeComponent();
        }
        private void Button_Clicked(object sender, EventArgs e)
        {
            try
            {
                float a = float.Parse(valor1.Text);
                float b = float.Parse(valor2.Text);
                float c;
                switch (operacion.SelectedItem.ToString())
                {
                    case "Suma":
                    {
                        c = a + b;
                        resultado.Text = a + "+" + b + "=" + c;
                        break;
                    }
                    case "Resta":
                    {
                        c = a - b;
                        resultado.Text = a + "-" + b + "=" + c;
                        break;
                    }
                }
            }
            catch
            {
                resultado.Text = "Datos Invalidos";
            }
        }
    }
}
```

2. Programa que utiliza un ListView para mostrar elementos que están almacenados en una colección.

Codigo 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"
             x:Class="ListView1.MainPage">

    <StackLayout>
        <!-- Place new controls here -->
        <Frame BackgroundColor="BlueViolet">
            <Label Text="ListView de frutas" HorizontalTextAlignment="Center"
                TextColor="White"/>
        </Frame>

        <Label Text="Nombre de la fruta:" Margin="15,0"/>
        <Entry x:Name="valor" Margin="15,0"/>

        <Label Text="URL de imagen de la fruta" Margin="15,0"/>
        <Entry x:Name="direccionurl" Margin="15,0"/>

        <Button Text="Agregar" CornerRadius="24" BackgroundColor="Orange"
            Clicked="Button_Clicked" Margin="15,1"/>

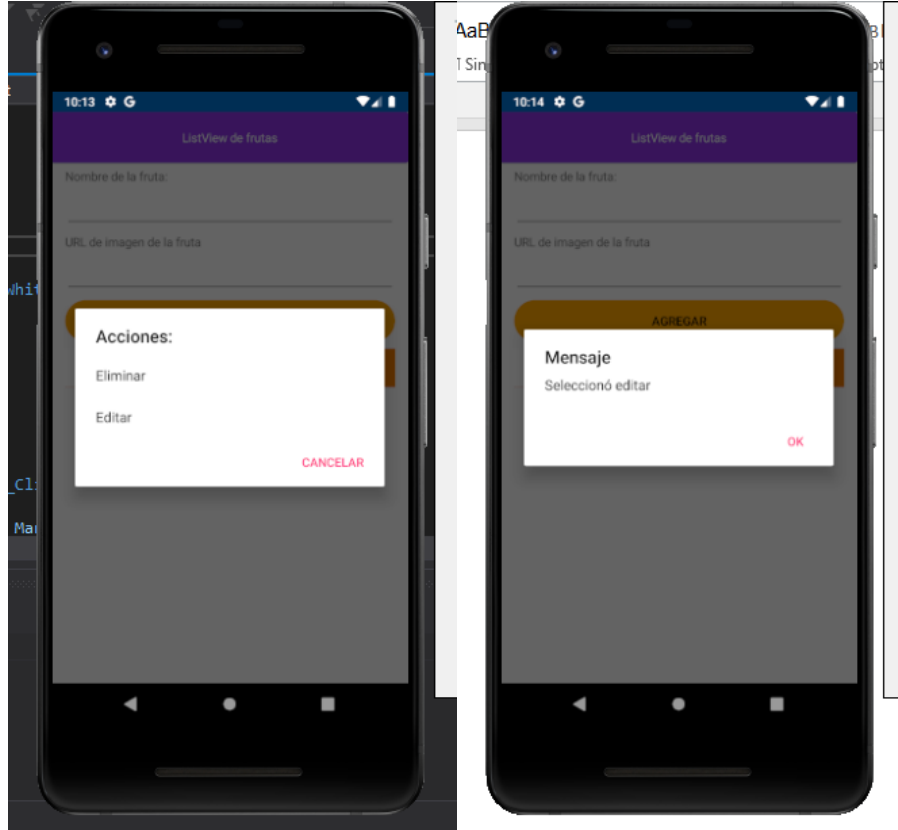
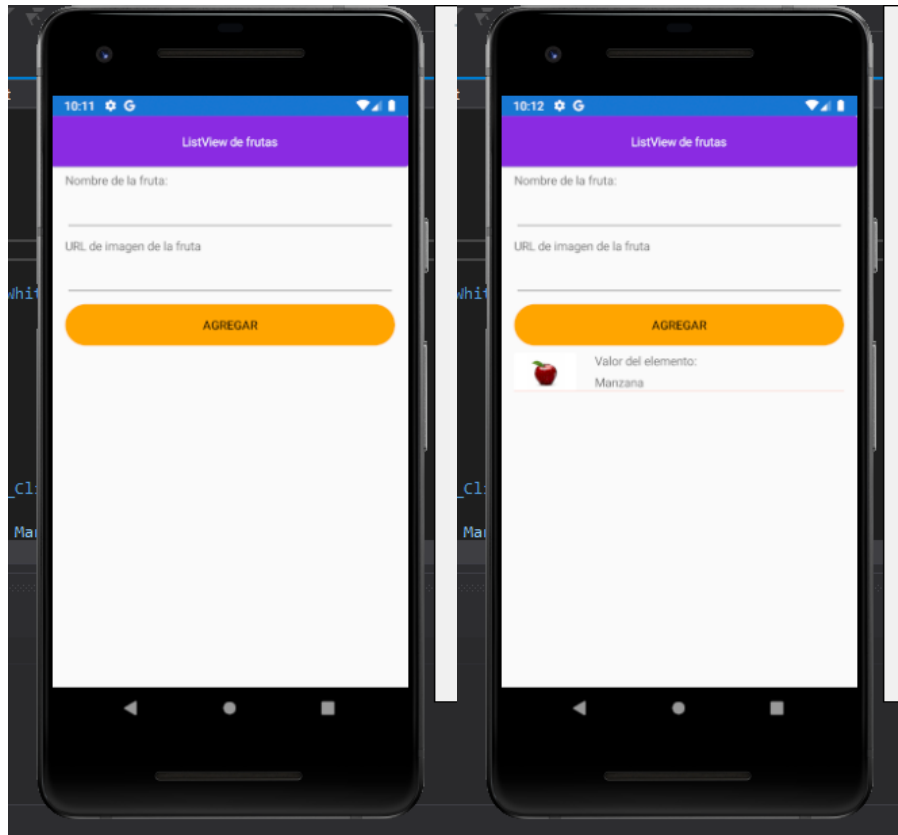
        <ListView x:Name="milista" ItemTapped="Milista_ItemTapped"
            SeparatorColor="Coral" Margin="15,1">
            <ListView.ItemTemplate>
                <DataTemplate>
                    <ViewCell>
                        <ViewCell.ContextActions>
                            <MenuItem Clicked="MenuItem_Mostrar" Text="Mostrar" />
                            <MenuItem Clicked="MenuItem_Borrar"
                                CommandParameter="{Binding Nombre}" Text="Borrar" IsDestructive="True" />
                        </ViewCell.ContextActions>

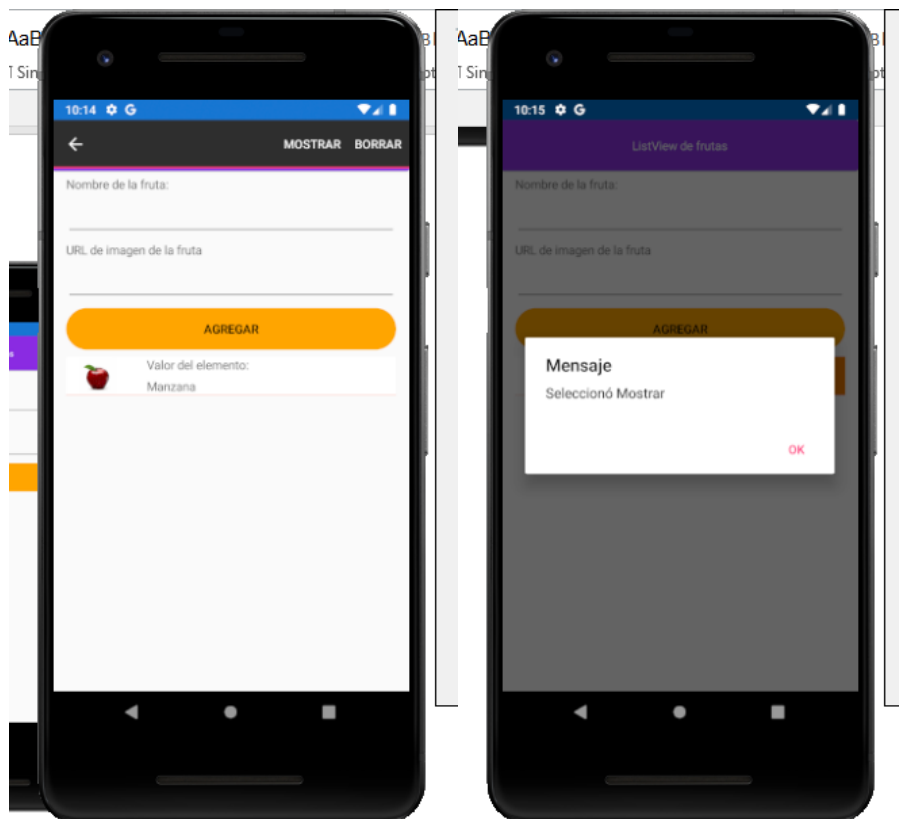
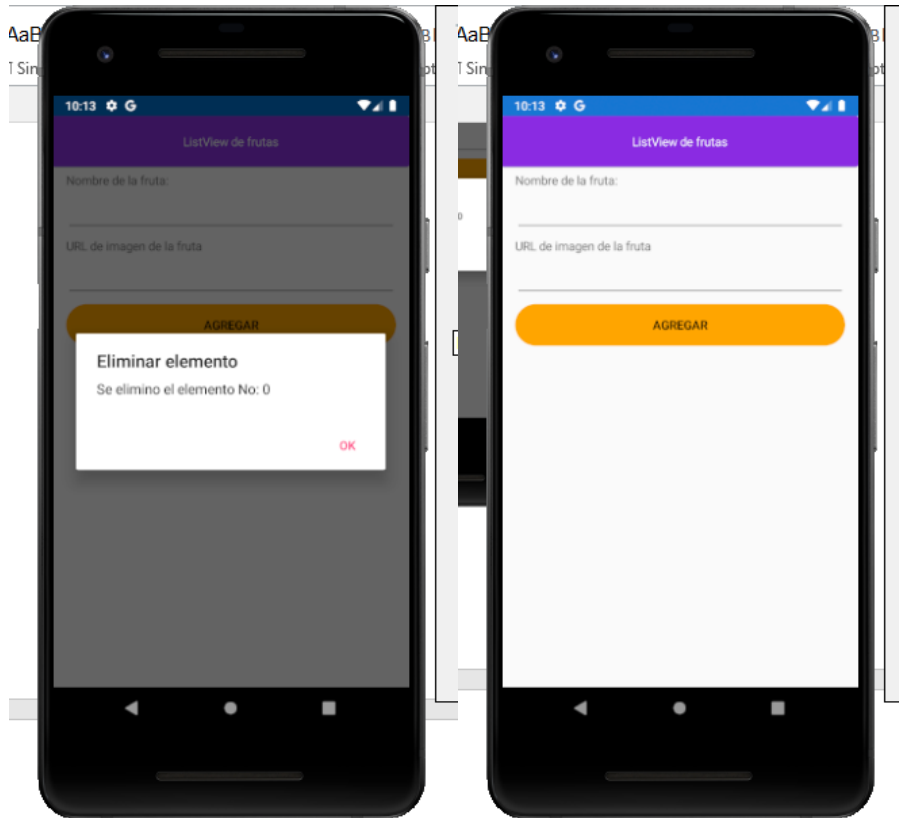
                        <StackLayout Orientation="Horizontal">
                            <Image Source="{Binding Url}"/>
                            <StackLayout Padding="15,0">
                                <Label Text="Valor del elemento:"/>
                                <Label Text="{Binding Nombre}" />
                            </StackLayout>
                        </StackLayout>
                    </ViewCell>
                </DataTemplate>
            </ListView.ItemTemplate>
        </ListView>
    </StackLayout>

</ContentPage>

```

Interfaz





Codigo C#

```
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;
using Xamarin.Forms.Internals;

namespace ListView1
{
    public partial class MainPage : ContentPage
    {
        public class Fruta
        {
            public string Nombre { get; set; }
            public string Url { get; set; }
        }
        ObservableCollection<Fruta> datos = new ObservableCollection<Fruta>();
        public MainPage()
        {
            InitializeComponent();
            milista.ItemsSource = datos;
        }
        //

        private void Button_Clicked(object sender, EventArgs e)
        {
            datos.Add(new Fruta { Nombre = valor.Text, Url = direccionurl.Text });
            valor.Text = "";
            direccionurl.Text = "";
        }
        private void MenuItem_Mostrar(object sender, EventArgs e)
        {
            DisplayAlert("Mensaje", "Seleccionó Mostrar", "ok");
        }
        private void MenuItem_Borrar(object sender, EventArgs e)
        {
            var mi = ((MenuItem)sender);
            DisplayAlert("Fruta seleccionada", "Fruta:" +
                mi.CommandParameter.ToString(), "ok");
        }

        }
        async private void Milista_ItemTapped(object sender, ItemTappedEventArgs e)
        {
            var myListView = (ListView)sender;
            var myItem = myListView.SelectedItem;
            int index = datos.IndexOf(myItem);
            string action = await DisplayActionSheet("Acciones:", "Cancelar",
                null, "Eliminar", "Editar");
            if (action == "Eliminar")
            {
                datos.RemoveAt(index);
            }
        }
    }
}
```



```

        await DisplayAlert("Eliminar elemento", "Se elimino el elemento No:
"
        + index, "OK");
        milista.ItemsSource = null;
        milista.ItemsSource = datos;
    }
    if (action == "Editar")
    {
        await DisplayAlert("Mensaje", "Seleccionó editar", "ok");
    }
}
}
}

```

- ¿Qué son las colecciones en C#?

Son un conjunto de clases diseñadas específicamente para agrupar objetos y llevar a cabo tareas con ellos.

- ¿Como se crea, añaden y eliminan los elementos en una colección de datos?

Para crear colección de datos

```
List<tipo> nombre = new List<tipo>();
```

Para añadir elementos

```
List<string> ListaColores = new List<string>();
```

```
ListaColores.Add ("Azul");
```

Para eliminar elementos

```
ListaColores.RemoveAt (index);
```