

# FIA/P GRADUAÇÃO

# **Xamarin**

## **•Aula anterior**

- Criando telas**
- Estratégia de navegação**
- Eventos e navegação entre telas**

## **•Aula de hoje**

- Layout (StackLayout, ScrollView)**
- Elementos Visuais**
- Listas**
- Acessando elementos**

# SISTEMAS DE INFORMAÇÃO

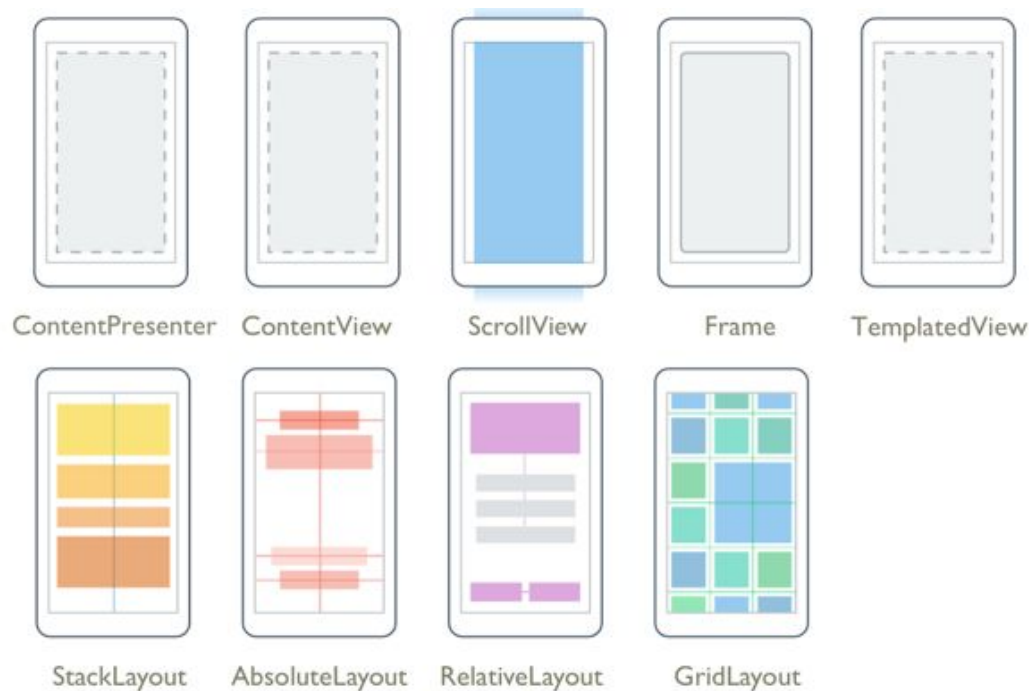
Xamarin

PROFº. FLÁVIO MORENI

# Layouts



- **Layouts** - são usados para organizar elementos da interface gráfico em uma estrutura lógica.



<https://developer.xamarin.com/guides/xamarin-forms/controls/layouts>

# Xamarin

- **StackLayout** - é um dos modelos de layout que posiciona os elementos em linhas orientados vertical ou horizontal.

- **Exemplo** - Xaml

```
➡ <StackLayout  
➡ VerticalOptions="CenterAndExpand">  
  
    <Button Text="Botão 1"  
        ➡ VerticalOptions="CenterAndExpand"  
        ➡ HeightRequest="120"/>  
  
    <Button Text="Botão 2"  
        VerticalOptions="CenterAndExpand"  
        HeightRequest="120"/>  
  
</StackLayout>
```



# Xamarin

- **StackLayout** - Posição horizontal.
  - **Exemplo** - Xaml

```
<StackLayout Orientation="Horizontal"
  ➔ HorizontalOptions="CenterAndExpand">

  <Button Text="Botão 1"
    VerticalOptions="CenterAndExpand"
    HeightRequest="120"/>

  <Button Text="Botão 2"
    VerticalOptions="CenterAndExpand"
    HeightRequest="120"/>

</StackLayout>
```

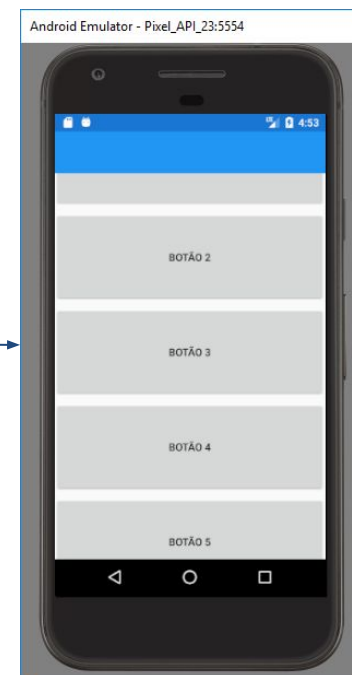
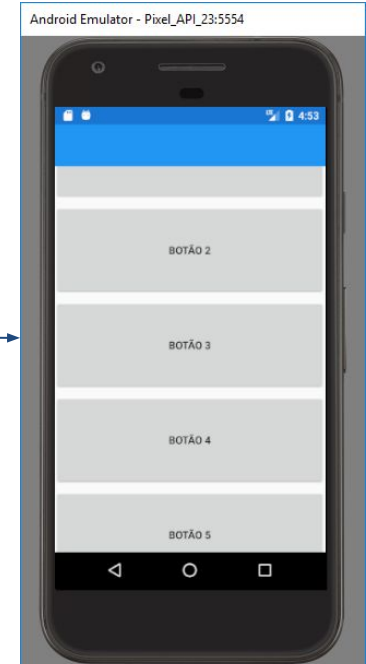


# Xamarin

- **ScrollView** - permite a rolagem para exibição de elementos posicionados fora da visão da página. Em caso de exibição do teclado o campo que será preenchido ficará visível ao usuário

- **Exemplo** - Xaml

```
➔ <ScrollView>  
    <StackLayout Orientation="Vertical">  
        <Button Text="Botão 1".../>  
        <Button Text="Botão 2".../>  
        <Button Text="Botão 3".../>  
        <Button Text="Botão 4".../>  
        <Button Text="Botão 5".../>  
        <Button Text="Botão 6".../>  
    </StackLayout>  
</ScrollView>
```







# Elementos Visuais

## (UI)



# Xamarin

- **Button** - componente visual que aceita toques do usuário.

➔ `<Button x:Name="Botao1"  
Text="Botão 1"  
VerticalOptions="CenterAndExpand"  
HeightRequest="120"  
BorderColor="Azure"  
FontSize="50"  
Image="icon.png"/>`



<https://developer.xamarin.com/api/type/Xamarin.Forms.Button/>

# Xamarin

- **Text (Label, Entry e Editor)** - componentes para exibição e captura de texto.

```
<Label Text="Nome" FontSize="Large"></Label>
```

```
➡ <Entry x:Name="EntryNome"  
    Placeholder="Antonio Coutinho"  
    FontSize="Large"></Entry>
```

```
➡ <Label Text="Descrição" FontSize="Large"></Label>
```

```
➡ <Editor x:Name="EditorDescricao"  
    HeightRequest="180"  
    FontSize="Large"  
    BackgroundColor="LightGray"></Editor>
```

```
<Button x:Name="BotaoGravar"  
    Text="Gravar"  
    FontSize="Large"></Button>
```



<https://developer.xamarin.com/guides/xamarin-forms/user-interface/text/>

# Xamarin

- **Slider** - componente para seleção linear de valores

```
<Label Text="Idade" FontSize="Large"></Label>  
➔ <Slider x:Name="SliderIdade"  
    Maximum="140"  
    Minimum="18"/>
```

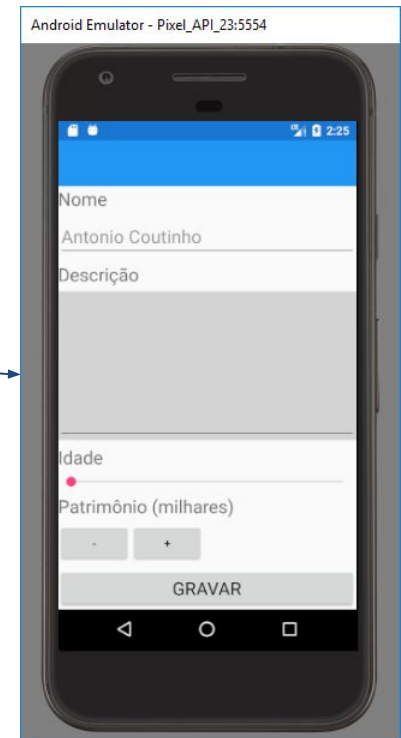


<https://developer.xamarin.com/api/type/Xamarin.Forms.Slider/>

# Xamarin

- **Stepper** - componente que permite aumentar ou valores de um determinado intervalo

```
<Label Text="Patrimônio (milhares)"
      FontSize="Large"></Label>
→ <Stepper x:Name="StepPatrimonio"
      Increment="1"
      Minimum="0"
      Maximum="10"
      Value="2"/>
```



<https://developer.xamarin.com/api/type/Xamarin.Forms.Stepper/>

# Xamarin

- **ListView** - apresenta coleções de dados de forma estruturada.

- XAML

```
➡ <ListView x:Name="ListViewPerfis"
    VerticalOptions="FillAndExpand"
    HorizontalOptions="FillAndExpand">
    ➡ <ListView.ItemTemplate>
        ➡ <DataTemplate>
            ➡ <ViewCell>
                <Label Text="{Binding .}"
                    FontSize="Large"/>
            </ViewCell>
        </DataTemplate>
    </ListView.ItemTemplate>
</ListView>
```

👉 <https://developer.xamarin.com/guides/xamarin-forms/user-interface/listview/>

# Xamarin

- **ListView** - apresenta coleções de dados de forma estruturada.

- Code.cs

```
public LoginPage ()  
{  
    InitializeComponent ();  
    → string[] valores = {  
        "Perfil Arrojado",  
        "Perfil Clássico",  
        "Perfil Arriscado" };  
    ListViewPerfis.ItemsSource = valores;  
}
```



<https://developer.xamarin.com/guides/xamarin-forms/user-interface/listview/>



## • **ListView** - código para copiar



- Code .cs

```
string[] valores = {"Perfil Arrojado", "Perfil Clássico", "Perfil Arriscado" };  
ListViewPerfis.ItemsSource = valores;
```

- XAML

```
<ListView x:Name="ListViewPerfis"  
    VerticalOptions="FillAndExpand"  
    HorizontalOptions="FillAndExpand">  
    <ListView.ItemTemplate>  
        <DataTemplate>  
            <ViewCell>  
                <Label Text="{Binding .}"  
                    FontSize="Large"/>  
            </ViewCell>  
        </DataTemplate>  
    </ListView.ItemTemplate>  
</ListView>
```





# Elementos Visuais

Acessando propriedades



• **Propriedades** - Abaixo seguem os elementos e as propriedades mais relevantes para nossa aplicação.

- **Entry** e **Editor** - propriedade **Text**
- **Slider** e **Stepper** - propriedade **Value**
- **ListView** - propriedade **SelectedItem**



# Xamarin

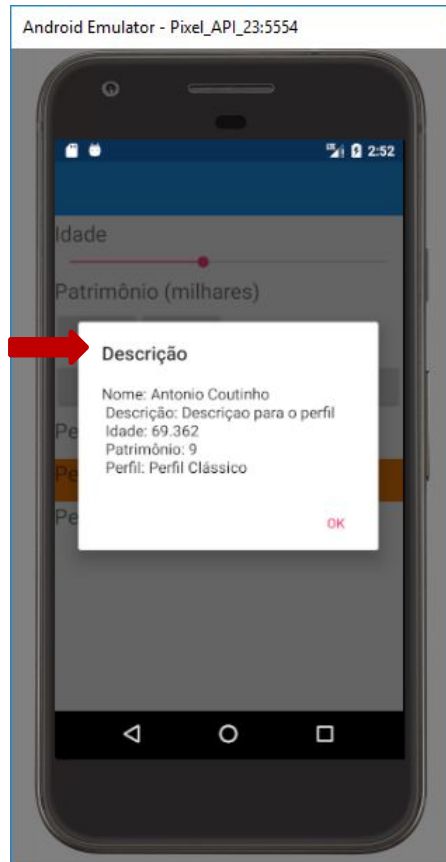
- **Propriedades** - Capturando os valores digitados e exibindo em uma mensagem de alerta.

```
var mensagemPerfil = "Nenhum perfil selecionado";  
→ if (ListViewPerfis.SelectedItem != null)  
    mensagemPerfil = ListViewPerfis.SelectedItem.ToString();  
  
var mensagem = "Nome: " + EntryNome.Text + "\n" +  
    "Descrição: " + EditorDescricao.Text + "\n" +  
    "Idade: " + SliderIdade.Value.ToString() + "\n " +  
    "Patrimônio: " + StepPatrimonio.Value.ToString() + "\n" +  
    "Perfil: " + mensagemPerfil + " \n ";  
  
→ await DisplayAlert("Descrição",mensagem,"Ok");
```



# Xamarin

## •Propriedades - resultado





# **Xamarin**

## **•Na aula de hoje**

- Layout (StackLayout, ScrollView)**
- Elementos Visuais**
- Listas**
- Acessando elementos**

## **•Próxima aula**

- Elementos Visuais (Cell)**
- ListView**
- Criando telas para aplicação**





## **Copyright © 2018 Profº. Flávio Moreni**

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).