

Macoratti.net Xamarin Android - Apresentando e usando o controle CheckBox



Neste artigo vou apresentar o controle (*widget*) **Checkbox** e mostrar como usar o controle em uma aplicação **Xamarin Android** usando o **Visual Studio Community 2015** e a linguagem **C#**.

Curso C# Vídeo Aulas
Do básico ao intermediário

Por um preço justo

O controle **CheckBox** (*ou caixa de seleção*) permite ao usuário selecionar um valor de uma lista de opções. O controle apresenta dois estados principais : **checked** (marcado) e **unchecked** (desmarcado).

Você geralmente utiliza este controle quando deseja apresentar ao usuário um grupo de opções selecionáveis que não sejam mutuamente exclusivas.

Vejamos a seguir um exemplo onde usamos o controle **CheckBox** para apresentar uma lista de linguagens que o usuário poderá selecionar.

A opção **Outras** quando selecionada dispara o evento **CheckedChange** que é tratado para exibir um **TextView** e um **EditText**, que estão ocultos, para que o usuário informe outra linguagem.

No exemplo iremos usar as seguintes propriedades/eventos do controle checkbox:

- **Click** - Evento que ocorre quando o controle foi clicado;
- **Checked** - Indica que o controle foi marcado;
- **Visibility** - Exibe/Oculta a View;
- **CheckedChange** - Evento disparado quando o controle é acionado;

Recursos usados:

- **Visual Studio Community 2015** ou **Xamarin Studio**
- **Xamarin**
- **Emulador Android virtual ou físico** ([veja como emular usando o Vysor](#))

Nota: Baixe e use a versão **Community 2015** do VS ela é grátis e é equivalente a versão **Professional**.

Criando o projeto no Visual Studio 2015 Community

Abra o **VS 2015 Community** e clique em **New Project**;

Selecione a linguagem **Visual C#** e o template **Android -> Blank App(Android)**

Informe o nome **App.UsandoCheckbox** e clique no botão **OK**;

Abra o arquivo **Main.axml** na pasta **Resources/layout** e no modo **Designer** inclua os seguintes controles

- 1 **TextView** (Large) - Text = **Selecione a linguagem**;
- 5 **CheckBox**
- 1 **TextView** (Medium)
- 1 **EditText** (Plain Text)

Abaixo vemos o leiaute no emulador do Xamarin exibindo a tela e ao lado o respectivo código XML gerado :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#003466"
    android:minWidth="25px"
    android:minHeight="25px">
    <TextView
        android:text="Selecione a linguagem :"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/textView1" />
    <CheckBox
        android:text="Visual C#"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkBox1"
        android:onClick="chkb_OnClick" />
    <CheckBox
        android:text="VB .NET"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkBox2"
        android:onClick="chkb_OnClick" />
    <CheckBox
        android:text="Android"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkBox3"
        android:onClick="chkb_OnClick" />
    <CheckBox
        android:text="Java"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/checkBox4
```

Os 4 primeiros controles **CheckBox** possuem o atributo :

android:onClick="chkb_OnClick"

Fizemos essa atribuição pois vamos tratar de forma declarativa o evento **OnClick** do controle.

```

        android:onClick="chkb_OnClick" />
<CheckBox
    android:text="Outras"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox5" />
<TextView
    android:text="Informe a linguagem : "
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/textView2" />
<EditText
    android:id="@+id/edittext"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content" />
</LinearLayout>

```

A seguir abra o arquivo **MainActivity.cs** e altere o código desse arquivo conforme abaixo:

```

using Android.App;
using Android.OS;
using Android.Views;
using Android.Widget;
using Java.Interop;

namespace App.UsandoCheckBox
{
    [Activity(Label = "App.UsandoCheckBox", MainLauncher = true, Icon = "@drawable/icon")]
    public class MainActivity : Activity
    {
        //declara as views
        CheckBox chkb1, chkb2, chkb3, chkb4, chkb5;
        TextView txtv2;
        EditText edtxt1;

        //define o tratamento do evento Click
        [Export("chkb_OnClick")]
        public void chkb_OnClick(View v)
        {
            //verifica o componente clicado
            switch (v.Id)
            {
                case Resource.Id.checkBox1:
                    Toast.MakeText(this, "Cliquei no checkbox1", ToastLength.Short).Show();
                    break;
                case Resource.Id.checkBox2:
                    Toast.MakeText(this, "Cliquei no checkbox2", ToastLength.Short).Show();
                    break;
                case Resource.Id.checkBox3:
                    Toast.MakeText(this, "Cliquei no checkbox3", ToastLength.Short).Show();
                    break;
                case Resource.Id.checkBox4:
                    Toast.MakeText(this, "Cliquei no checkbox4", ToastLength.Short).Show();
                    break;
            }
        }

        protected override void OnCreate(Bundle bundle)
        {
            base.OnCreate(bundle);

            // Set our view from the "main" layout resource
            SetContentView(Resource.Layout.Main);

            chkb1 = FindViewById<CheckBox>(Resource.Id.checkBox1);
            chkb2 = FindViewById<CheckBox>(Resource.Id.checkBox2);
            chkb3 = FindViewById<CheckBox>(Resource.Id.checkBox3);
            chkb4 = FindViewById<CheckBox>(Resource.Id.checkBox4);
            chkb5 = FindViewById<CheckBox>(Resource.Id.checkBox5);
            //textView
            txtv2 = FindViewById<TextView>(Resource.Id.textView2);
            txtv2.Visibility = ViewStates.Invisible;
            //edittext
            edtxt1 = FindViewById<EditText>(Resource.Id.edittext);
            edtxt1.Visibility = ViewStates.Invisible;

            //evento do checkbox
            chkb5.CheckedChange += Chkb5_CheckedChange;
        }

        //tratamento do evento CheckedChange
        private void Chkb5_CheckedChange(object sender, CompoundButton.CheckedChangeEventArgs e)
        {
            CheckBox checkbox = (CheckBox)sender;

            //oculta/exibe o textView e o EditText
            if (checkbox.Checked)
            {
                txtv2.Visibility = ViewStates.Visible;
            }
        }
    }
}

```

```

        edtxt1.Visibility = ViewStates.Visible;
    }
    else
    {
        txtv2.Visibility = ViewStates.Invisible;
        edtxt1.Visibility = ViewStates.Invisible;
    }
}
}
}

```

Neste código estamos fazendo o seguinte:

1- Declarando variáveis para tratar as instâncias das Views definidas no arquivo **Main.axml**:

```

//declara as views
CheckBox chkb1, chkb2, chkb3, chkb4, chkb5;
TextView txtv2;
EditText edtxt1;

```

2- Tratando o evento **OnClick** do controle **CheckBox** declarado no arquivo **Main.axml**. Para isso precisamos incluir uma referência à biblioteca **Mono.Android.Export** no projeto e incluir o namespace **using Java.Interop**:

```

[Export("chkb_OnClick")]
public void chkb_OnClick(View v)
{
    ....
}

```

Para cada controle que for clicado exibimos uma mensagem.

3- Ocultamos os controles **TextView** e **EditText** usando o atributo **Visibility** das views:

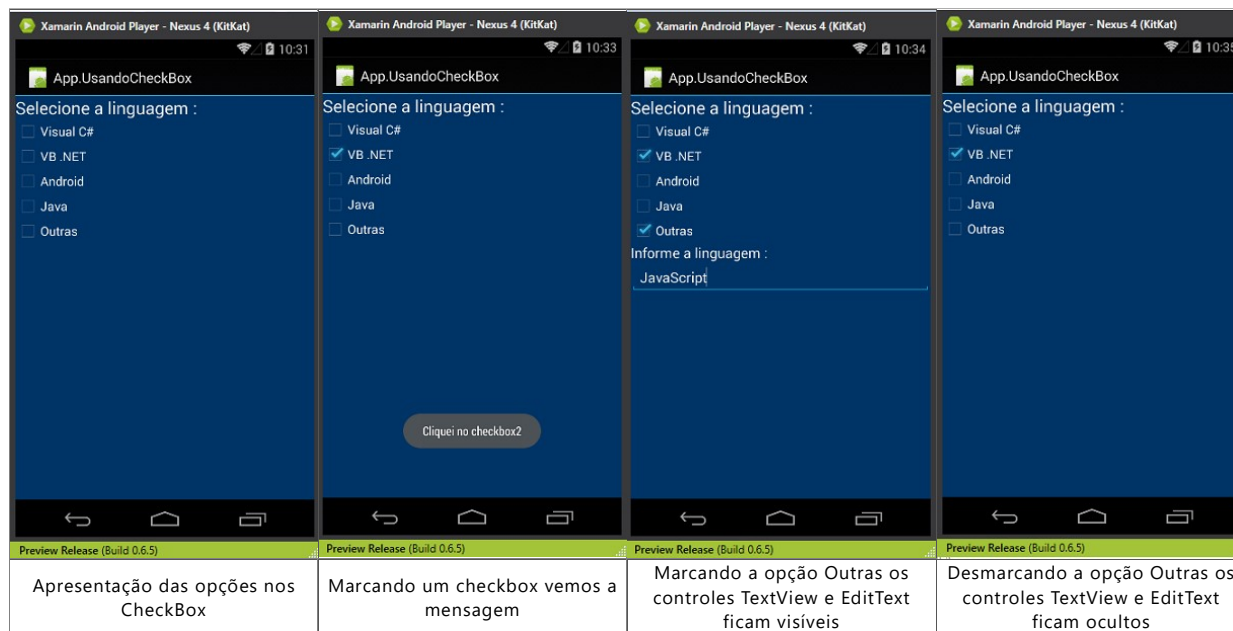
```

//textView
txtv2 = FindViewById<TextView>(Resource.Id.textView2);
txtv2.Visibility = ViewStates.Invisible;
//edittext
edtxt1 = FindViewById<EditText>(Resource.Id.edittxt);
edtxt1.Visibility = ViewStates.Invisible;
.....

```

4- Tratamos o evento **CheckedChange** do **checkbox5 (Outras)** e se o ele estiver marcado exibimos os controles **TextView** e **EditText**. Caso contrário ocultamos esses controles.

Executando o código acima teremos o seguinte resultado :



Pegue o projeto completo aqui : [App.UsandoCheckBox.zip](#) (sem as referências)

Deitar-me faz em verdes pastos, guia-me mansamente a águas tranqüilas.Refrigera a minha alma; guia-me pelas veredas da justiça, por amor do seu nome.

Salmos 23:2,3

[Veja os Destaques e novidades do SUPER DVD Visual Basic \(sempre atualizado\) : clique e confira !](#)

Quer migrar para o VB .NET ?

- Veja mais sistemas completos para a plataforma .NET no [Super DVD .NET](#) , confira...

- [Curso Básico VB .NET - Vídeo Aulas](#)

Quer aprender C# ??

- Chegou o [Super DVD C#](#) com exclusivo material de suporte e vídeo aulas com curso básico sobre C#.

- [Curso C# Basico - Video Aulas](#)

Quer aprender os conceitos da Programação Orientada a objetos ?

- [Curso Fundamentos da Programação Orientada a Objetos com VB .NET](#) NEW

Quer aprender o gerar relatórios com o ReportViewer no VS 2013 ?

- [Curso - Gerando Relatórios com o ReportViewer no VS 2013 - Vídeo Aulas](#) NEW

Referências:

- [Seção VB .NET do Site Macoratti.net](#)
- [Super DVD .NET - A sua porta de entrada na plataforma .NET](#)
- [Super DVD Vídeo Aulas - Vídeo Aula sobre VB .NET, ASP .NET e C#](#)
- [Super DVD C# - Recursos de aprendizagens e vídeo aulas para C#](#)
- [Seção C# do site Macoratti.net](#)
- [Seção ASP .NET do site Macoratti .net](#)
- [Curso Básico VB .NET - Vídeo Aulas](#)
- [Curso C# Básico - Vídeo Aulas](#)
- [Curso Fundamentos da Programação Orientada a Objetos com VB .NET](#) NEW
- [Macoratti .net | Facebook](#)
- [macoratti - YouTube](#)
- [Jose C Macoratti \(@macorati\) | Twitter](#)
- [Xamarin - Desenvolvimento Multiplataforma com C# ... - Macoratti.net](#)
- [Xamarin - Apresentando Xamarin.Forms - Macoratti.net](#)
- [Xamarin.Forms - Olá Mundo - Criando sua primeira ... - Macoratti.net](#)
- [Xamarin.Forms - Olá Mundo - Anatomia da aplicação - Macoratti.net](#)
- <https://developer.xamarin.com/api/type/Android.App.AlertDialog/>
- [Xamarin Android - Tratando eventos de forma declarativa](#)

[José Carlos Macoratti](#)