

## Macoratti.net Xamarin Android - Criando uma tela de apresentação

---



Neste tutorial vou mostrar como criar uma tela de apresentação em uma aplicação **Android** usando [Visual Studio](#) com **Xamarin**.

Criar uma tela de apresentação para sua aplicação no Xamarin Android pode ser uma tarefa bem simples, basta escolher uma imagem, definir um arquivo **XML de estilo** e criar uma **Activity** para exibir a tela de apresentação.

Vejamos isso na prática usando o Visual Studio com Xamarin e a linguagem C#.

### Recursos usados:

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

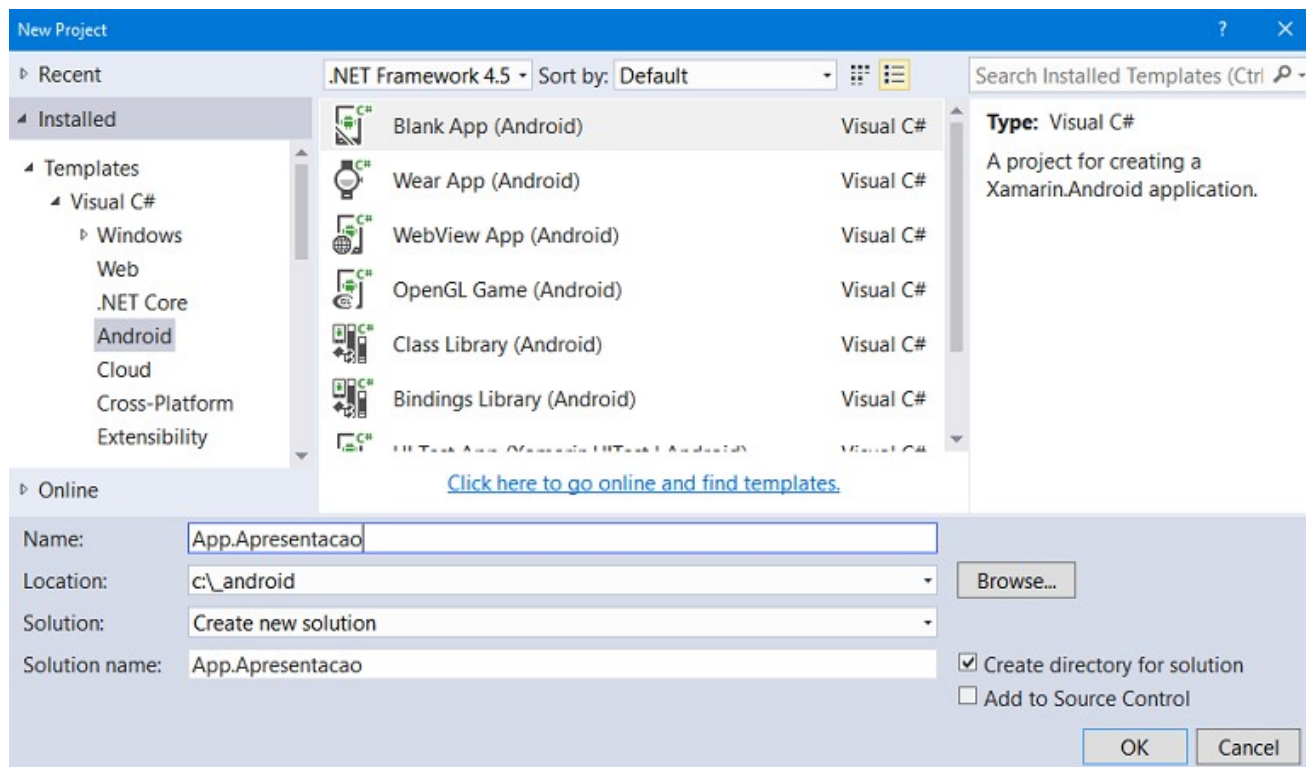
**Nota: Baixe e use a versão Community 2015 do VS ela é grátis e é equivalente a versão Professional.**

### Criando o projeto no Visual Studio 2015 Community

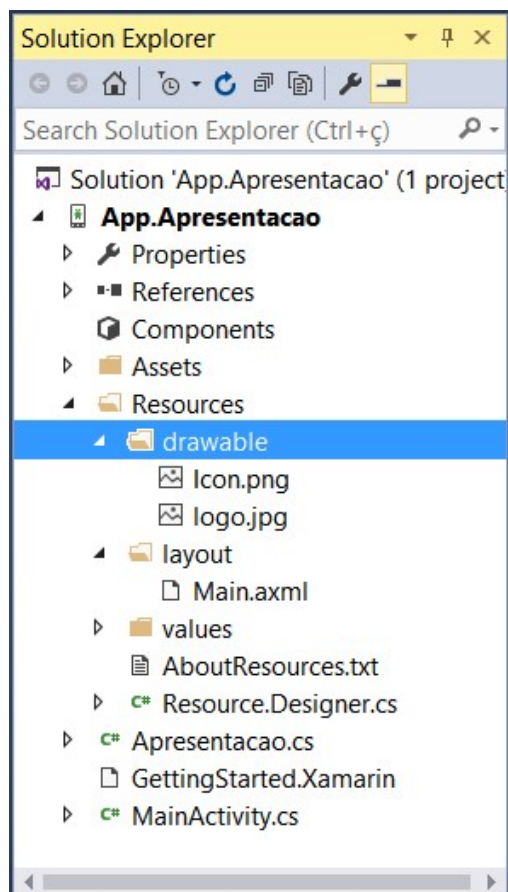
Abra o [VS 2015 Community](#) e clique em **New Project**;

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

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



Será criada uma solução com a seguinte estrutura:



- **Properties** - Contém o arquivo [AndroidManifest.xml](#) que descreve as funcionalidades e requisitos da sua aplicação Android, e o arquivo [AssemblyInfo.cs](#) contém informação sobre o projeto como número de versão e build.

- **References** - Contém as bibliotecas [Mono.Android](#), [System.Core](#) e todas as bibliotecas usadas no seu projeto;

- **Components** - Contém componentes de terceiros ou desenvolvidos por você usados no seu projeto.

A maioria dos componentes está disponíveis diretamente do **Xamarin Component Store** e são **free** (*não todos*) e prontos para serem usados; (Para incluir um componente clique com o botão direito sobre **Components** e a seguir em [Get More Components](#));

- **Assets e Resources** - Contém arquivos que não são código, como imagens, sons, arquivos XML e qualquer outro recurso que sua aplicação for usar. Os arquivos externos colocados na pasta **Assets** são facilmente acessíveis em tempo de execução através do [Asset Manager](#).

Já os arquivos colocados na pasta **Resources** precisam ser declarados e mantidos em uma lista com os **IDs** dos recursos que você deseja usar em tempo de execução.

De forma geral, todas as imagens, ícones, sons e outros arquivos externos são colocados na pasta **Resources** enquanto que dicionários e arquivos XML são postos na pasta **Assets**;

Na subpasta **layout** temos os arquivos [.axml](#) que definem as [views](#) usadas no projeto;

Na subpasta **values** temos o arquivo [Strings.xml](#) onde definimos as strings usadas no projeto;

**Nota** : A pasta **Drawable** contém recursos como imagens png, jpg, etc., usadas no aplicativo. Ela contém múltiplas pastas específicas para cada resolução possível em uma aplicação Android. Numa aplicação típica Android você vai acabar encontrando as pastas: [Drawable-LDPI](#), [Drawable-mdpi](#), [Drawable-hdpi](#), [Drawable-xhdpi](#), [Drawable-xxhdpi](#), etc.

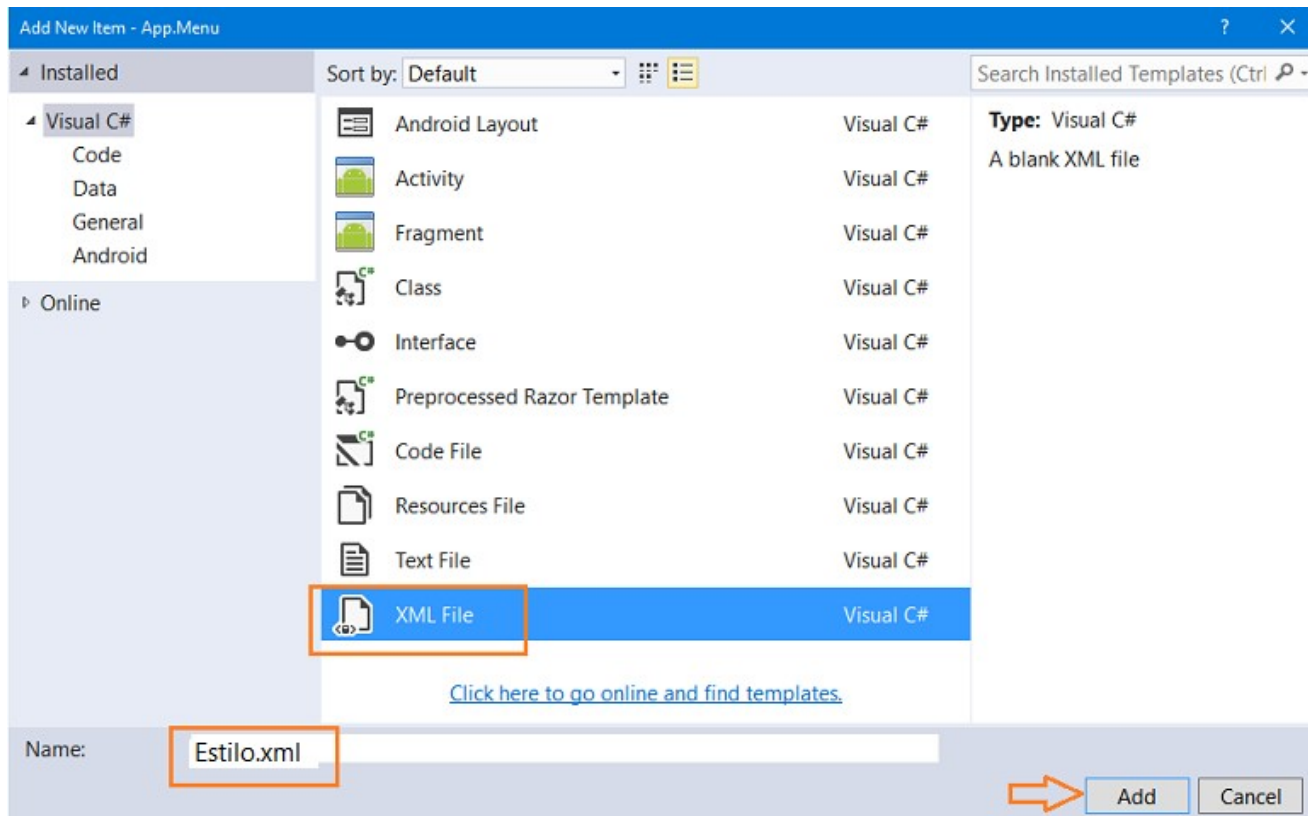
A primeira coisa a fazer é escolher uma imagem que vamos exibir na tela de apresentação e incluí-la na pasta **drawable**.

Eu estou escolhendo uma imagem com tamanho de **480x800** com o nome **logo.jpg** vista a seguir:



Vamos continuar criando um novo arquivo XML chamado **estilo.xml** na pasta **values** onde iremos definir o nosso tema e a imagem que iremos exibir na tela de apresentação da aplicação.

Clique com o botão direito do mouse sobre a pasta **menu** e depois clique em **Add New Item**;



Clique no botão **Add** e a seguir inclua o código abaixo no arquivo XML:

```
<?xml version="1.0" encoding="utf-8" ?>
<resources>
  <style name="macoratti.Theme" parent="android:Theme">
    <item name="android:windowBackground">
      @drawable/logo
    </item>
    <item name="android:windowNoTitle">true</item>
  </style>
</resources>
```

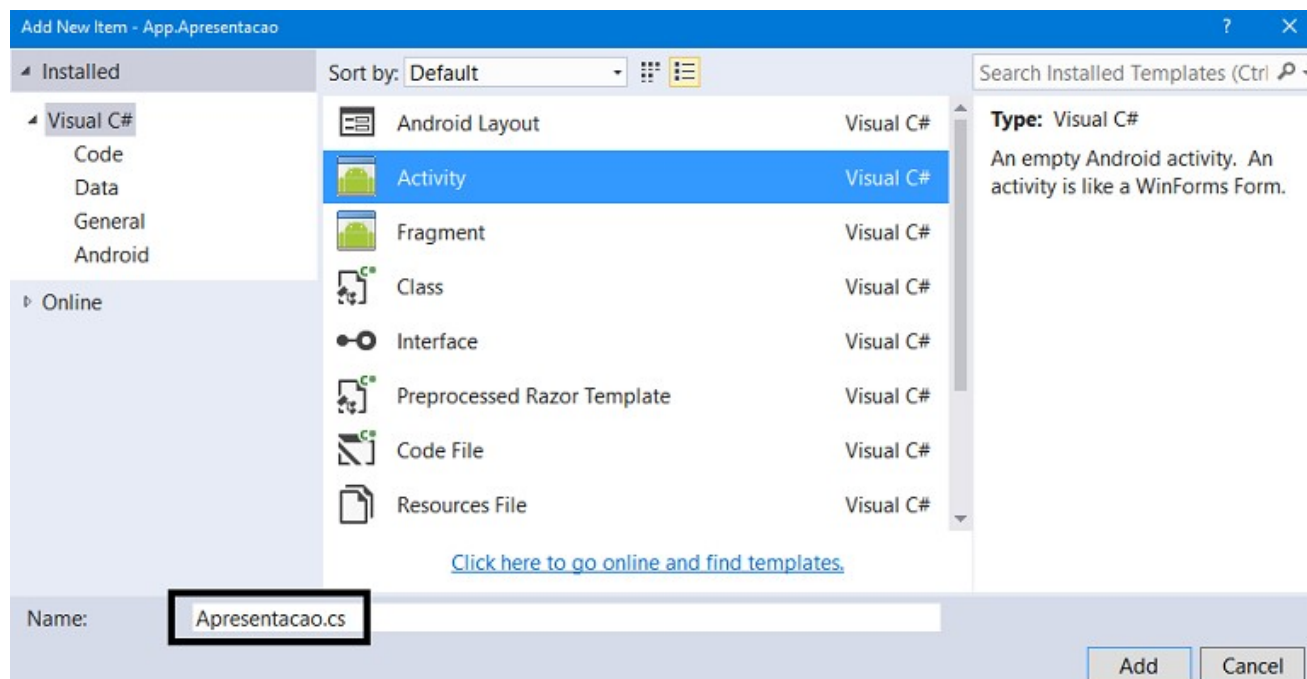
No arquivo XML definimos os seguinte itens:

- nome do estilo : **macoratti.Theme**
- a janela de fundo : **android:windowBackground**
- a imagem a ser usada : **@drawable/logo**
- a definição que a janela não terá um título : **android:windowNoTitle**

Vamos incluir uma nova Activity no projeto.

No menu **Project -> Add New Item**;

Selecione o template **Activity** e informe o nome **Apresentacao.cs** e clique no botão **Add**;



Veja como deve ficar o código do arquivo **Apresentacao.cs**:

```
using Android.App;
using Android.OS;
using System.Threading;

namespace App.Apresentacao
{
    [Activity(Theme = "@style/macoratti.Theme", MainLauncher = true, NoHistory = true)]
    public class Apresentacao : Activity
    {
        protected override void onCreate(Bundle savedInstanceState)
        {
            base.OnCreate(savedInstanceState);
            Thread.Sleep(4000);
            StartActivity(typeof(MainActivity));
        }
    }
}
```

Neste código temos o atributo **MainLauncher= true** o que indica que a atividade atual é o ponto de entrada da aplicação.

Temos mais duas definições :

- 1- **Theme = "@style/macoratti.Theme"** - que indica o tema que estamos usando e que foi definido no arquivo **estilo.xml**;
- 2- **NoHistory = true** - é usado para evitar que o usuário use o botão voltar e navegue pelo seu histórico de atividades;

Depois de dar uma pausa de 4s na aplicação : **Thread.Sleep(4000);**

Chamamos a Activity **MainActivity** : **StartActivity**(typeof(**MainActivity**));

Para concluir basta definir o código da Activity **MainActivity.cs** conforme abaixo:

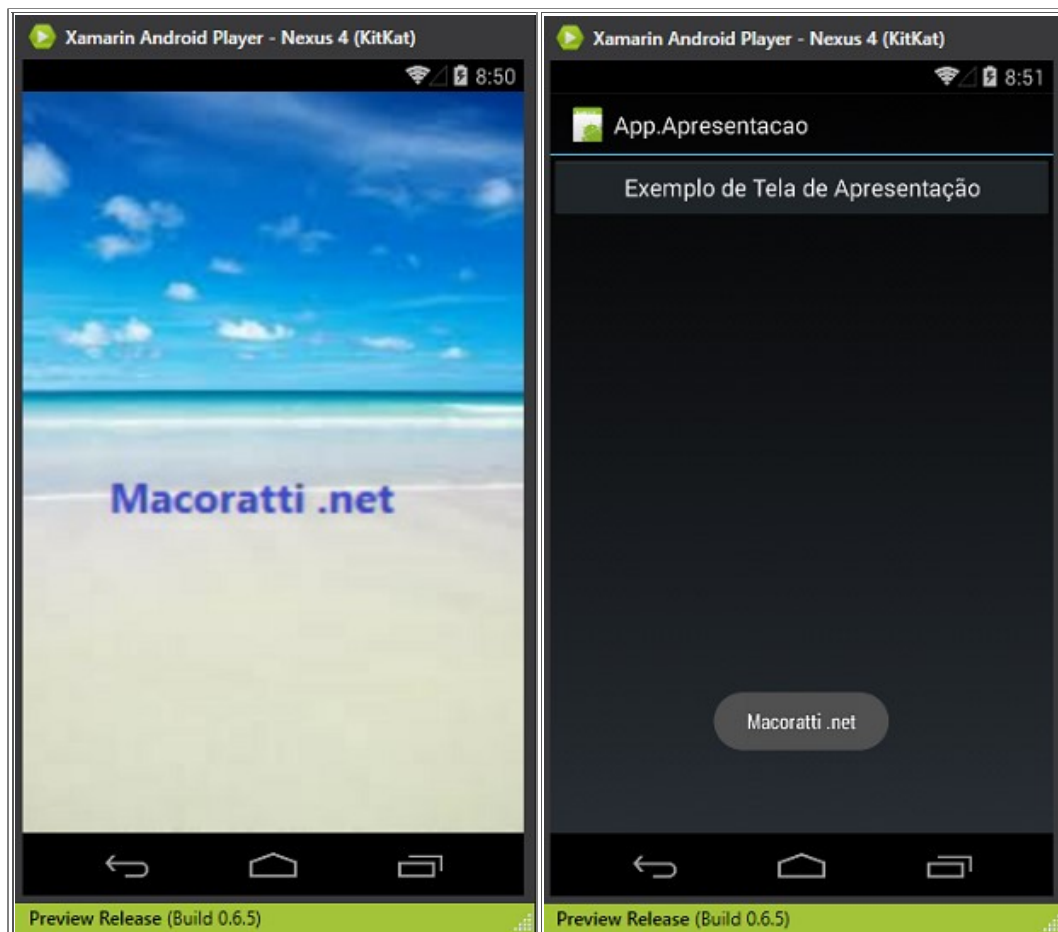
```
using Android.App;
using Android.OS;
using Android.Widget;

namespace App.Apresentacao
{
    [Activity(Label = "App.Apresentacao", Icon = "@drawable/icon")]
    public class MainActivity : Activity
    {
        protected override void OnCreate(Bundle bundle)
        {
            base.OnCreate(bundle);
            SetContentView(Resource.Layout.Main);

            Button button = FindViewById<Button>(Resource.Id.MyButton);
            button.Click += delegate
            {
                Toast.MakeText(this, "Macoratti .net", ToastLength.Short).Show();
            };
        }
    }
}
```

Observe que não estamos usando o atributo **MainLauncher** para esta Activity.

Executando novamente o projeto teremos na carga da Activity principal teremos a exibição da nossa tela de apresentação:



Aguarde mais artigos sobre a criação de aplicações com Xamarin.Android.

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

**(Disse Jesus aos seus discípulos) : "Se vós fôsseis do mundo, o mundo amaria o que era seu, mas porque não sois do mundo, antes eu vos escolhi do mundo, por isso é que o mundo vos odeia."**

**João 15:19**

[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/>

---

José Carlos Macoratti