



Neste artigo vou iniciar a utilização do **Xamarin.Essentials** mostrando como verificar a conexão com a internet em uma aplicação Xamarin Forms.



**Curso de Xamarin Forms Vídeo Aulas**  
Desenvolva para Android, iOS e Windows Phone

Se você costuma usar plugins para incluir funcionalidades em suas aplicações Xamarin não pode deixar de conhecer o **Xamarin.Essentials**.

Eu apresentei o **Xamarin.Essentials** neste artigo : [Xamarin Forms - Introdução ao Xamarin.Essentials](#)

Hoje eu vou mostrar como podemos verificar a conexão com a internet usando o **Xamarin.Essentials**.

A informação sobre conectividade usando o Xamarin.Essentials não é 100% precisa; você pode receber a informação de que a rede esta disponível mas o acesso total à web não esta.

Devido ao funcionamento de conectividade em cada plataforma o resultado obtido só pode garantir que uma conexão está disponível. Assim, o dispositivo pode estar conectado a uma rede **Wi-Fi**, mas o **roteador** estar desconectado da internet.

A informação do acesso à rede pode ser:

- **Internet-Local** – acesso Local e internet;
- **ConstrainedInternet** – acesso limitado à internet;
- **Local** – Local apenas acesso à rede;
- **None** – nenhuma conectividade está disponível;
- **Unknown** – não é possível determinar a conectividade com a internet;

Além disso você pode verificar que tipo de perfil de conexão o dispositivo esta usando. Ele que descreve o tipo de conexão que o dispositivo estiver usando ativamente:

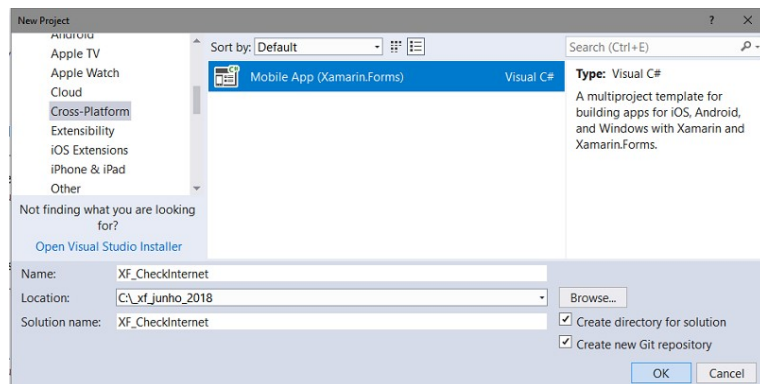
<b>Bluetooth</b>	0	Conexão bluetooth
<b>Cellular</b>	1	Conexão mobile/cellular
<b>Ethernet</b>	2	Conexão ethernet
<b>Other</b>	5	Tipo de conexão não conhecido
<b>WiFi</b>	4	Conexão WiFi
<b>WiMAX</b>	3	Conexão WiMAX

#### Recursos Usados

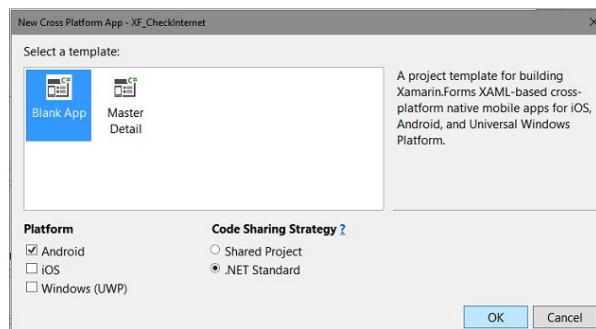
- [Visual Studio 2017 Community](#)
- [Xamarin.Essentials](#)

## Criando o projeto Xamarin Forms

Abra o VS 2017 Community e clique em **New Project** e a seguir escolha **Cross Platform** -> **Mobile App (Xamarin.Forms)** e informe o nome **XF\_CheckInternet**:



A seguir selecione a Plataforma, eu marquei somente **Android**, e escolha a estratégia de compartilhamento que será **.NET Standard**.



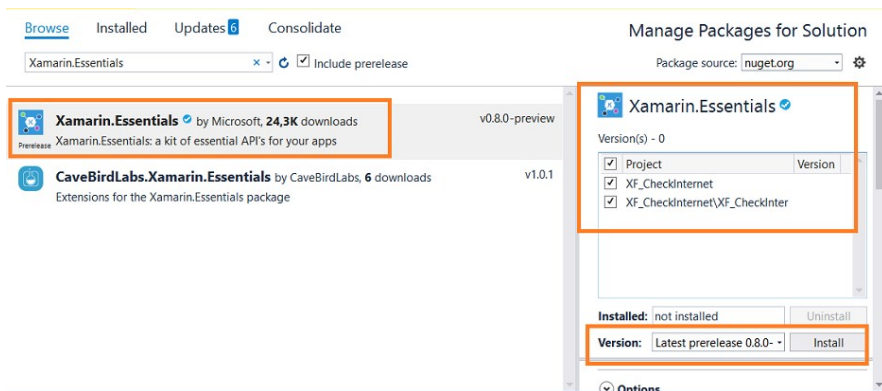
Clique no botão **OK**.

Pronto nosso projeto já esta criado. (Verifique a necessidade de atualizar a versão do Xamarin Forms no seu projeto. Atualmente(06/2018) a versão mais atual é a 3.1.0)

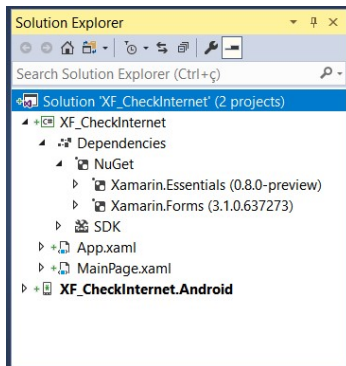
### Incluindo a referência ao Plugin no projeto

Acesse o **Manage Packages for Solution** via menu **Tools** e na guia **Browse** localize e instale o plugin **Xamarin.Essentials** em todos os projetos:

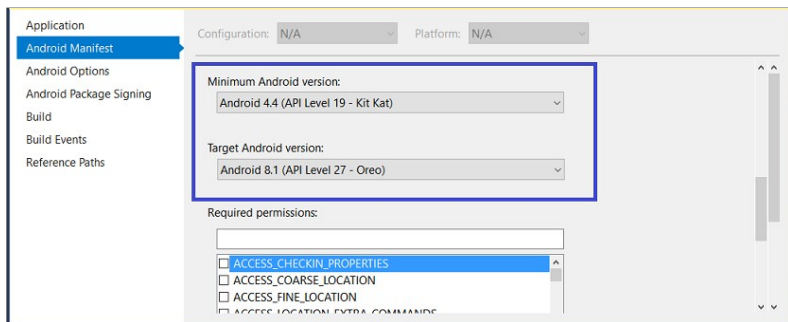
Como o plugin ainda esta em **pré-release** para poder instalar o pacote via Nuget marque a opção : **include prerelease**



Ao final seu projeto deverá conter as referências aos pacotes :

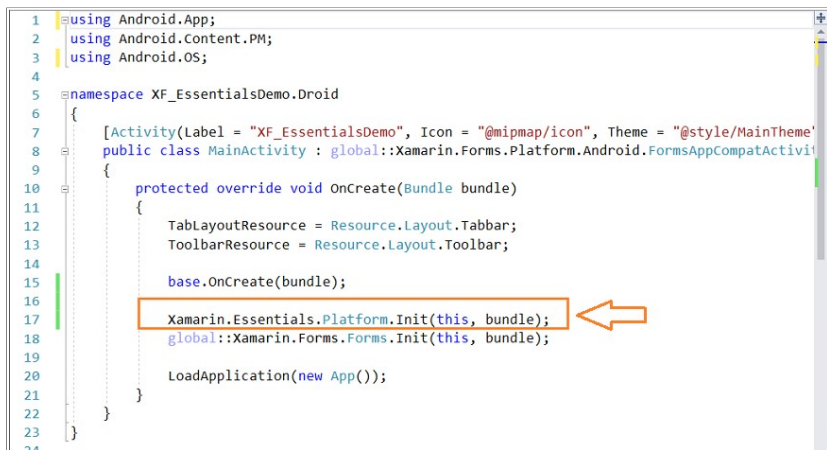


Lembrando que o **Xamarin.Essentials** dá suporte a uma versão mínima do **Android do 4.4**, correspondente ao nível de **API 19**, mas a versão do Android de destino de compilação deve ser 8.1, correspondente ao nível 27 da API. Então, nas propriedades do seu projeto Android as configurações mínimas devem estar definidas conforme mostra a figura abaixo :



Após criar o seu projeto Xamarin Forms, no projeto Android **MainLauncher** ou em qualquer **Activity** que inicia o projeto o **Xamarin.Essentials** deve ser inicializado no método **OnCreate**.

Assim no arquivo **MainActivity** você deve incluir a linha de código mostrada na figura abaixo:



Para poder lidar com permissões em tempo de execução no Android, o plugin deve receber um **OnRequestPermissionsResult** que deve ser incluído nas Activities.

Abaixo o código necessário incluído na **MainActivity** do projeto Android:

```

7  [Activity(Label = "XF_EssentialsDemo", Icon = "@mipmap/icon", Theme = "@style/MainTheme",
8  MainLauncher = true, ConfigurationChanges = ConfigChanges.ScreenSize | ConfigChanges.Orientation)]
9  public class MainActivity : global::Xamarin.Forms.Platform.Android.FormsAppCompatActivity
10 {
11     protected override void OnCreate(Bundle bundle)
12     {
13         TabLayoutResource = Resource.Layout.Tabbar;
14         ToolbarResource = Resource.Layout.Toolbar;
15
16         base.OnCreate(bundle);
17
18         Xamarin.Essentials.Platform.Init(this, bundle);
19         global::Xamarin.Forms.Forms.Init(this, bundle);
20
21         LoadApplication(new App());
22     }
23
24     public override void OnRequestPermissionsResult(int requestCode, string[] permissions, [GeneratedEnum] Permission[] grantResults)
25     {
26         Xamarin.Essentials.Platform.OnRequestPermissionsResult(requestCode, permissions, grantResults);
27         base.OnRequestPermissionsResult(requestCode, permissions, grantResults);
28     }
29 }
30

```

Não esqueça também de definir no projeto Android as permissões : **INTERNET** e **ACCESS\_NETWORK\_STATE**

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="1" android:versionName="1.0" package="com.companynome" android:installLocation="auto">
    <uses-sdk android:minSdkVersion="19" android:targetSdkVersion="27" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <application android:label="XF_CheckInternet.Android"></application>
</manifest>

```

Na pasta **Resources/drawable** do projeto Android eu incluí duas imagens para exibir no projeto: **mac.jpg** e **xamessential.png** que são opcionais.

## Definindo o código da MainPage.xaml e do code-behind

Abra o arquivo **MainPage.xaml** e inclua no código abaixo:

```

<?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:XF_CheckInternet"
    x:Class="XF_CheckInternet.MainPage">

    <StackLayout HorizontalOptions="Center" VerticalOptions="Start">
        <Image Margin="0,50,0,0" x:Name="imgBanner" Source="mac.jpg" ></Image>
        <Image Margin="0,0,0,10" x:Name="imgXamarinEssential" Source="xamessential.png" ></Image>
        <Label Margin="0,0,0,10" FontAttributes="Bold" FontSize="Medium" TextColor="#CA6F1E" HorizontalTextAlignment="Center" Text="Informação de Conectividade"></Label>
        <Label HorizontalOptions="Center" FontAttributes="Bold" FontSize="Medium" x:Name="lblNetworkStatus"></Label>
        <Label HorizontalOptions="Center" FontAttributes="Bold" FontSize="Medium" x:Name="lblNetworkPerfil"></Label>
    </StackLayout>

</ContentPage>

```

Agora para concluir inclua o código a seguir no arquivo **MainPage.xaml.cs** :

```

using System.Linq;
using Xamarin.Essentials;
using Xamarin.Forms;

namespace XF_CheckInternet
{
    public partial class MainPage : ContentPage
    {
        public MainPage()
        {
            InitializeComponent();

            var internet = Connectivity.NetworkAccess;
            var perfis = Connectivity.Profiles;

            //verifica o acesso à internet
            if (internet == NetworkAccess.Internet)
            {
                lblNetworkStatus.Text = "A internet esta Disponível";
            }
            else
            {
                lblNetworkStatus.Text = "A internet esta Indisponível";
            }

            //verifica o perfil de conectividade
            if (perfis.Contains(ConnectionProfile.WiFi))
            {
                lblNetworkPerfil.Text = perfis.FirstOrDefault().ToString();
            }
            else
            {
                lblNetworkPerfil.Text = perfis.FirstOrDefault().ToString();
            }
        }
    }
}

```

```
}
```

Executando o projeto iremos obter o seguinte resultado:



Pegue o código do projeto compartilhado aqui : [XF\\_CheckInternet.zip](#)

'(Disse Jesus) - Nisto todos conhecerão que sois meus discípulos, se vos amardes uns aos outros.'  
**João 13:35**

[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](#)

Gostou ?  [Compartilhe no Facebook](#)  [Compartilhe no Twitter](#)

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#](#)
- [Seção C# do site Macoratti.net](#)
- [Super DVD C#](#)
- [Super DVD Visual Basic](#)
- [Curso Básico VB .NET - Vídeo Aulas](#)
- [Curso C# Básico - Vídeo Aulas](#)
- [Xamarin Android - Apresentando o controle ProgressBar - Macoratti.net](#)
- [Xamarin Android - Usando o serviço de Alarme - Macoratti.net](#)
- [Xamarin.Forms - Principais Recursos - Macoratti.net](#)
- [Xamarin.Forms - Previsão do Tempo - Macoratti.net](#)
- [Xamarin Android - Aplicando animações - Macoratti.net](#)
- [Xamarin Android - Usando a classe Timer - Macoratti.net](#)
- [Curso de Xamarin.Forms - Macoratti.net](#)
- [Xamarin.Forms - Usando a view ActivityIndicator - Macoratti.net](#)
- [Xamarin.Forms - MVVM - Macoratti.net](#)
- [Xamarin.Forms - Usando a Câmera - Macoratti.net](#)

[José Carlos Macoratti](#)