



Neste tutorial vou mostrar como implementar o **login** usando um banco de dados **SQLite** no Xamarin Android com **VS 2015 Community** e a linguagem **C#**.

Curso C# Vídeo Aulas
Do básico ao intermediário

Por um preço justo

O **SQLite** é uma biblioteca de software escrito em C desenvolvido por **D. Richard Hipp** em 2000 originalmente como parte de um contrato com a Marinha dos EUA.

Hoje, ele consiste de mais ou menos 184.000 linhas de código e é de domínio público, por isso pode ser usado por qualquer pessoa.

Embora você possa encontrar referências ao **SQLite** como sendo um SGBD, ele é, estritamente falando, apenas esta biblioteca.

O recipiente no qual a biblioteca é compilada (*uma classe, um framework, ou um SGBD de pleno direito*) fornece as funcionalidades de DBMS maiores.

Como o **SQLite** não requer administração, ele funciona bem em dispositivos que devem operar sem o suporte de um especialista. Por isso ele é muito usado em celulares, televisores, consoles de jogos, câmeras, relógios, utensílios de cozinha, termostatos, automóveis, máquinas-ferramentas, aviões, sensores remotos, zangões, dispositivos médicos, e robôs: a "*internet das coisas*".

Assim, um banco de dados SQL completo com várias **tabelas, índices, triggers e views**, está contido em um único arquivo de disco.

O formato de arquivo de banco de dados é multiplataforma - você pode copiar livremente um banco de dados entre os sistemas de 32 bits e de 64 bits ou entre arquiteturas de *big-endian* e *little-endian*.

A versão atual do SQLite é o **SQLite 3**.

Neste tutorial eu vou criar uma aplicação **Xamarin Android** que usar o banco de dados **SQLite** para realizar a autenticação e registro de usuário em uma operação de login.

Vamos criar 3 arquivos de layout que serão as views da interface com o usuário:

- **Main.axml**
- **NovoUsuario.axml**
- **Login.axml**

Vamos criar 3 Activity que irão gerenciar a respectiva view:

- **MainActivity (Main)**
- **RegistrarActivity (NovoUsuario)**
- **LoginActivity (Login)**

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**;

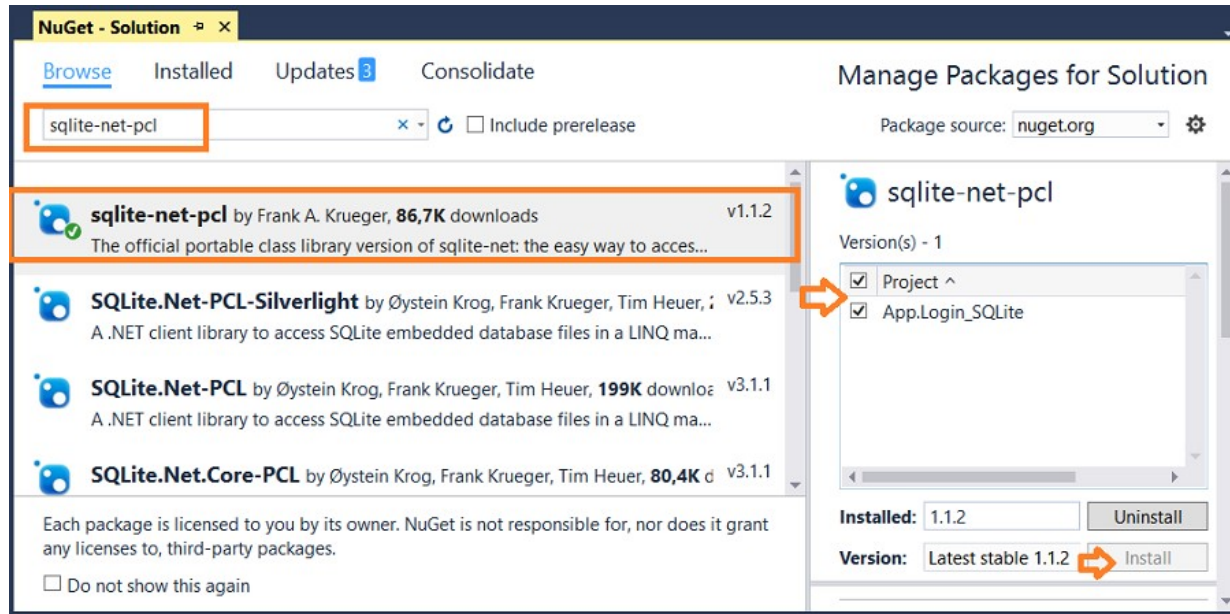
Selecione a linguagem Visual C# e o template **Android -> Blank App(Android)**

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

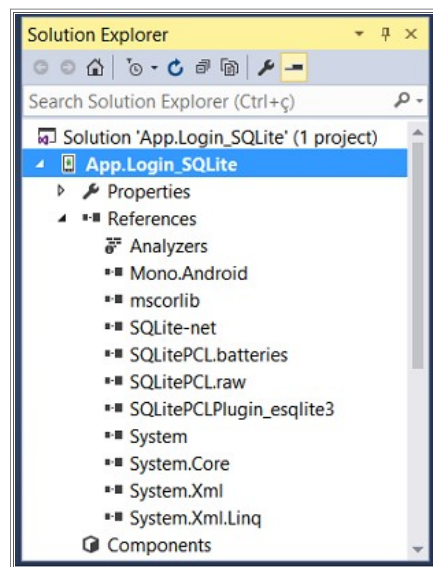
Referenciando o SQLite no projeto via Nuget

No menu **Tools** clique em **Nuget Package Manager** e a seguir em **Manage Nuget Packages for Solution**;

Digite **sqlite-net-pcl** e selecione o pacote, o projeto e clique no botão **Install**:



Ao final você deverá ver os pacotes instalados indicados em **References**:



Assim estamos prontos para usar os recursos do SQLite no projeto.

Definindo o modelo de domínio : a classe Login

Clique com o botão direito sobre a pasta **Resources** e a seguir em **Add Folder** e informe o nome **Model**.

Selecione a pasta **Model** e no menu **Project** clique em **Add Class** e informe o nome **Login.cs**.

A seguir defina o código da classe **Login** conforme o código abaixo:

```
using SQLite;

namespace
```

```

App.Login_SQLite.Resources.Model
{
    public class Login
    {
        [PrimaryKey, AutoIncrement]
        public int id { get; set; }

        [MaxLength(25)]
        public string usuario { get; set; }

        [MaxLength(15)]
        public string senha { get; set; }
    }
}

```

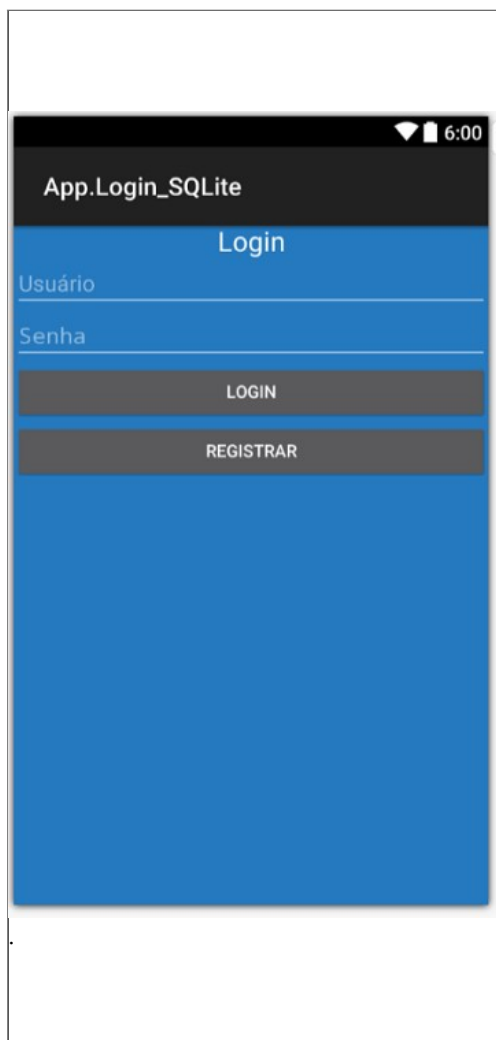
O atributo o **[PrimaryKey, AutoIncrement]** torna o campo **id** uma chave primária autoincremental.

Definindo a interface com o usuário no arquivo Main.xml

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

- 1 TextView - txtviewLogin
- 2 EditText - txtUsuario, txtSenha
- 2 Button - btnLogin e btnRegistrar

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

	<pre> <?xml version="1.0" encoding="utf-8"?> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" android:orientation="vertical" android:background="#2579BF" android:layout_width="match_parent" android:layout_height="match_parent"> <TextView android:text="Login" android:gravity="center" android:textAppearance="?android:attr/textAppearanceLarge" android:layout_width="match_parent" android:layout_height="wrap_content" android:id="@+id/txtviewLogin" /> <EditText android:hint="Usuário" android:layout_width="match_parent" android:layout_height="wrap_content" android:id="@+id/txtUsuario" /> <EditText android:hint="Senha" android:layout_width="match_parent" android:layout_height="wrap_content" android:inputType="textPassword" android:id="@+id/txtSenha" /> <Button android:text="Login" android:layout_width="match_parent" android:layout_height="wrap_content" android:id="@+id/btnLogin" /> <Button android:text="Registrar" android:layout_width="match_parent" android:layout_height="wrap_content" android:id="@+id/btnRegistrar" /> </LinearLayout> </pre>
--	---

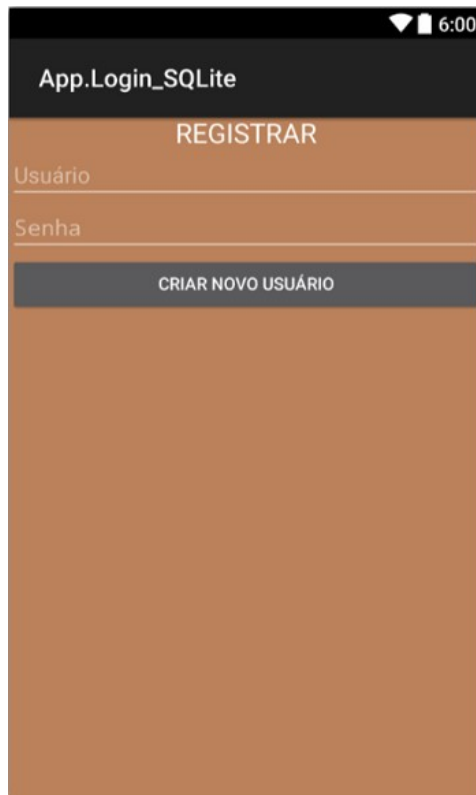
Definindo a interface do usuário para Registrar um novo usuário : NovoUsuario.xml

Clique com o botão direito do mouse sobre a pasta **Resources/layout** e no menu **Project** clique em **Add New Item**;

Selecione o template **Android Layout** e informe o nome **NovoUsuario.xml** e clique em **Add**;

A seguir vamos incluir a partir da **ToolBox** os seguintes controles :

- 1 TextView : txtviewRegistrar
- 2 EditText : txtNovoUsuario, txtSenhaNovoUsuario
- 1 Button : btnRegistrar



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:background="#ba815a"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:text="REGISTRAR"
        android:gravity="center"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/txtviewRegistrar" />
    <EditText
        android:hint="Usuário"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/txtNovoUsuario" />
    <EditText
        android:hint="Senha"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:id="@+id/txtSenhaNovoUsuario" />
    <Button
        android:text="CRIAR NOVO USUÁRIO"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/btnRegistrar" />
</LinearLayout>
```

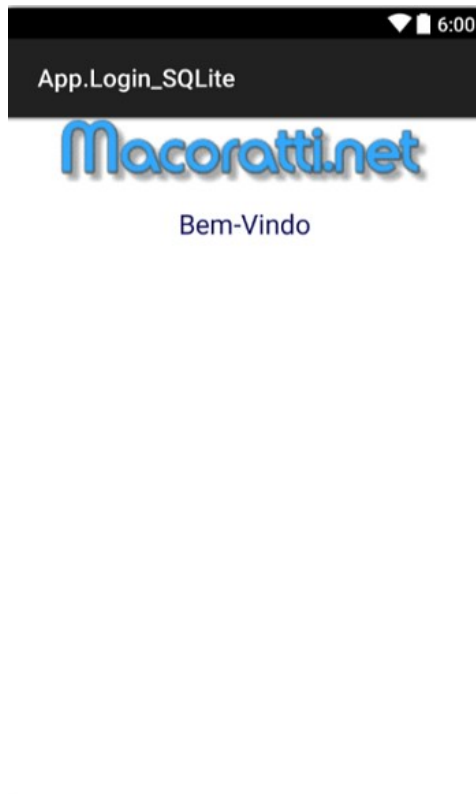
Definindo a interface do usuário para o login com sucesso : Login.xml

Clique com o botão direito do mouse sobre a pasta **Resources/layout** e no menu **Project** clique em **Add New Item**;

Selecione o template **Android Layout** e informe o nome **Login.xml** e clique em **Add**;

A seguir vamos incluir a partir da **ToolBox** os seguintes controles :

- 1 ImageView : imagem: maco1b.jpg
- 1 TextView : txtTextoLogin



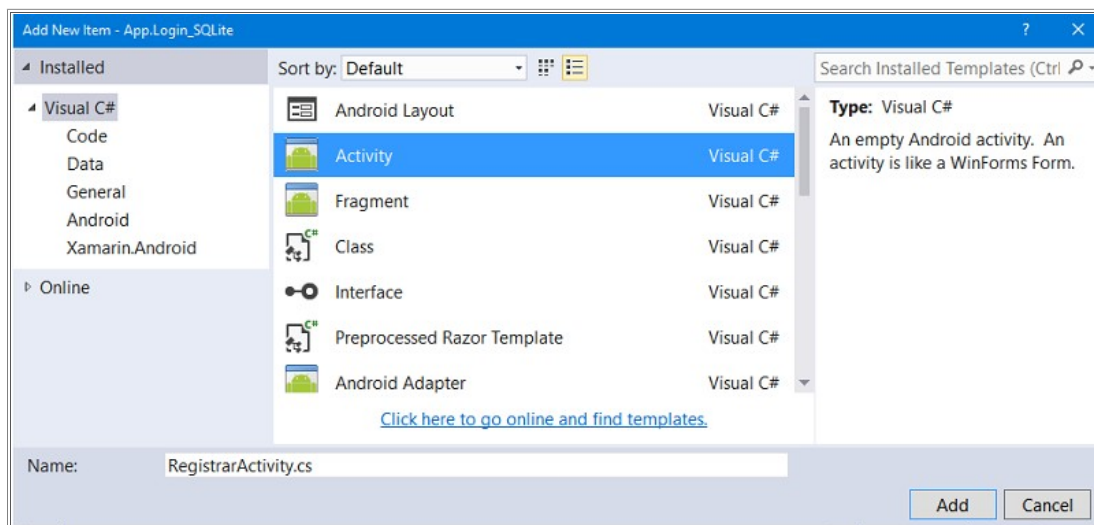
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ffffff"
    android:minWidth="25px"
    android:minHeight="25px">
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginBottom="20dp"
        android:src="@drawable/maco1b" />
    <TextView
        android:gravity="center"
        android:text="Bem-Vindo"
        android:textColor="#0e0e58"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/txtTextoLogin" />
</LinearLayout>
```

Criando as activities para gerenciar as views NovoUsuario.xml e Login.xml

Vamos agora criar duas novas **Activities** em nossa aplicação que irão gerenciar as respectivas views : **NovoUsuario.xml** e **Login.xml**.

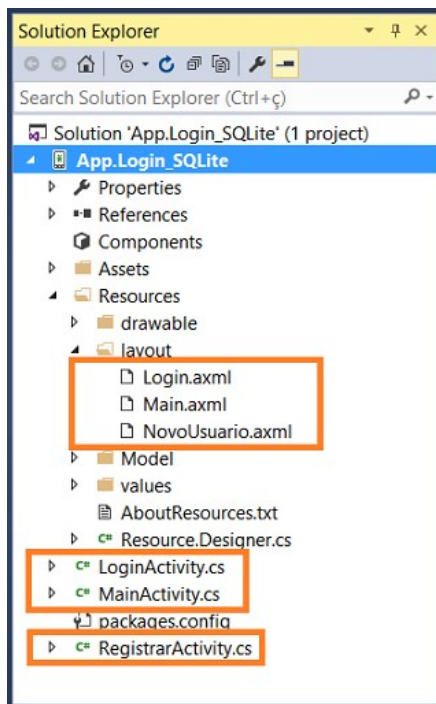
No menu **Project** clique em **Add New Item**;

A seguir selecione o template **Android** e **Activity** informando o nome **RegistrarActivity.cs** e clique no botão **Add**;



Repita o procedimento acima e crie a Activity **LoginActivity.cs**.

Neste momento nosso projeto apresenta 3 arquivos de layout e 3 Activities.



Na [segunda parte do artigo](#) veremos a implementação do código em cada Activity.

Jesus Cristo é o mesmo, ontem, e hoje, e eternamente.

[Hebreus 13:8](#)

[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 - Video 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 \(@macoratti\) | 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](#)
- [Seção Mobile/Xamarin do site Macoratti .net](#)

[José Carlos Macoratti](#)