



Neste artigo vou mostrar como fazer realizar a paginação de dados no GridView eu uma aplicação [Xamarin Android](#) usando o Visual Studio 2017 e a linguagem C#.



Curso de Xamarin Forms Vídeo Aulas

Desenvolva para Android, iOS e Windows Phone

Neste artigo, veremos como implementar uma simples paginação de dados em um **GridView** de forma que ao chegar na última página o botão **Próximo** será desativado e quando estivermos na primeira página o botão **Anterior** será desativado.

Lembrando que a GridView é um [ViewGroup](#) que exibe itens em uma grade bidimensional e rolável. Os itens da grade são inseridos automaticamente no layout usando um [ListAdapter](#).

De forma resumida vamos aprender o seguinte:

- [Pagar dados em um GridView usando um ArrayList](#)
- [Preencher o GridView com dados usando uma classe](#)
- [Navegar nos dados](#)
- [Usar a view GridView](#)

Recursos usados:

- [Visual Studio Community 2017](#) ou [Xamarin Studio](#)
- [Xamarin](#)

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

Criando o projeto no VS Community 2017

Abra o [VS 2017 Community](#) e clique em **New Project**;

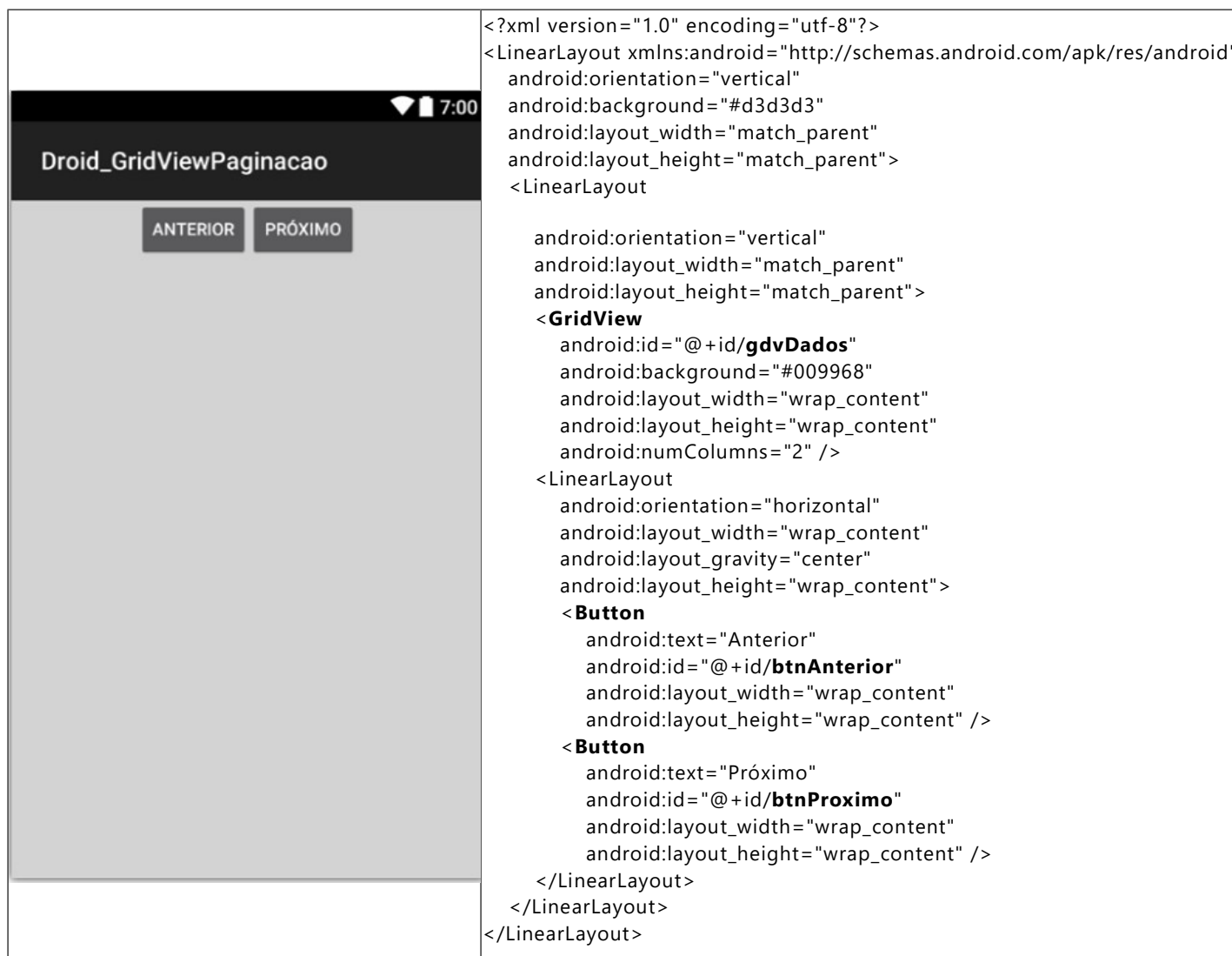
Selecione a linguagem Visual C# e o template **Android -> Blank App(Android)**

Informe o nome **Droid_GridViewPaginacao** e clique no botão **OK**;

Abra o arquivo **Main.axml** na pasta **Resources/layout** e no modo **Designer** e a seguir inclua a partir da ToolBox os seguintes controles:

- [GridView - gdvDados](#)
- [Button - btnProximo](#)
- [Button - btnAnterior](#)

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



Agora vamos definir o código no arquivo **MainActivity.cs** vinculado a nossa view **Main.axml**.

Definindo a Navegação

Abra o arquivo **MainActivity.cs** e altere o código desse arquivo conforme abaixo:

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

namespace Droid_GridViewPaginacao
{
    [Activity(Label = "Droid_GridViewPaginacao", MainLauncher = true, Icon = "@drawable/icon")]
    public class MainActivity : Activity
    {
        private GridView gv;
        private Button proxBtn, anteBtn;
        private readonly Paginador pag = new Paginador();
        private const int totalPaginas = Paginador.TOTAL_NUM_ITEMS / Paginador.ITEMS_POR_PAGINA;
        private int paginaAtual = 0;

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

            // Set our view from the "main" layout resource
```

```
SetContentView (Resource.Layout.Main);
this.initializeViews();
//Vincula a primeira pagina
gv.Adapter = new ArrayAdapter(this, Android.Resource.Layout.SimpleListItem1, pag.GerarPagina(paginaAtual));
}

private void initializeViews()
{
    //Referencia as views
    gv = FindViewById<GridView>(Resource.Id.gdvDados);
    proxBtn = FindViewById<Button>(Resource.Id.btnProximo);
    anteBtn = FindViewById<Button>(Resource.Id.btnAnterior);

    anteBtn.Enabled = false;
    //Eventos Clicks dos botões
    proxBtn.Click += ProxBtn_Click;
    anteBtn.Click += AnteBtn_Click;
}

//Anterior
private void AnteBtn_Click(object sender, System.EventArgs e)
{
    paginaAtual -= 1;
    gv.Adapter = new ArrayAdapter(this, Android.Resource.Layout.SimpleListItem1, pag.GerarPagina(paginaAtual));
    AlternarBotoes();
}

//Proximo
private void ProxBtn_Click(object sender, System.EventArgs e)
{
    paginaAtual += 1;
    gv.Adapter = new ArrayAdapter(this, Android.Resource.Layout.SimpleListItem1, pag.GerarPagina(paginaAtual));
    AlternarBotoes();
}

/*
 * Alterna o estado dos botões
 */
private void AlternarBotoes()
{
    if (paginaAtual == totalPaginas)
    {
        proxBtn.Enabled = false;
        anteBtn.Enabled = true;
    }
    else
    if (paginaAtual == 0)
    {
        anteBtn.Enabled = false;
        proxBtn.Enabled = true;
    }
    else
    if (paginaAtual >= 1 && paginaAtual <= 5)
    {
        proxBtn.Enabled = true;
        anteBtn.Enabled = true;
    }
}
}
```

O código é bem simples e já está comentado mas basicamente definimos os controles a serem usados , preenchemos o GridView e implementamos o método **GerarPagina** na classe **Paginador**.

No exemplo estamos usando um [ArrayAdapter](#) e estamos usando um leiaute de linha existente chamado **SimpleListItem1**, que representa uma única linha de texto, para definir a aparência do GridView. Este layout de item contém um único **TextView** permitindo exibir uma única linha de texto.

Nota: *Existem diversos layouts de itens de lista incorporados ao Xamarin.Android como : **SimpleListItem2**, **TwoLineListItem**, **ActivityListItem**, **SimpleListItem2**, **TestListItem**, etc.*

```
gv.Adapter = new ArrayAdapter<this, Android.Resource.Layout.SimpleListItem1,
pag.GerarPagina(paginaAtual))
```

Os argumentos usados são:

- O primeiro argumento é **this** : é o contexto da aplicação;
- O segundo argumento é o leiaute definido no arquivo XML que possui o **TextView** para cada item do array. Estamos usando : **SimpleListItem1**
- O terceiro argumento é o **array** de strings que será gerado para preencher o gridview;

Código da classe Paginador

A seguir temos o código da classe **Paginador()** psso

```
using System.Collections;

namespace Droid_GridViewPaginacao
{
    public class Paginador
    {
        //CONSTANTES
        public const int TOTAL_NUM_ITEMS = 54;
        public const int ITEMS_POR_PAGINA = 10;
        public const int ITEMS_RESTANTES = TOTAL_NUM_ITEMS % ITEMS_POR_PAGINA;
        public const int ULTIMA_PAGINA = TOTAL_NUM_ITEMS / ITEMS_POR_PAGINA;
        /*
        * GERA UMA ÚNICA PÁGINA DE DADOS
        */
        public ArrayList GerarPagina(int paginaAtual)
        {
            int itemInicial = paginaAtual * ITEMS_POR_PAGINA + 1;
            const int numDados = ITEMS_POR_PAGINA;

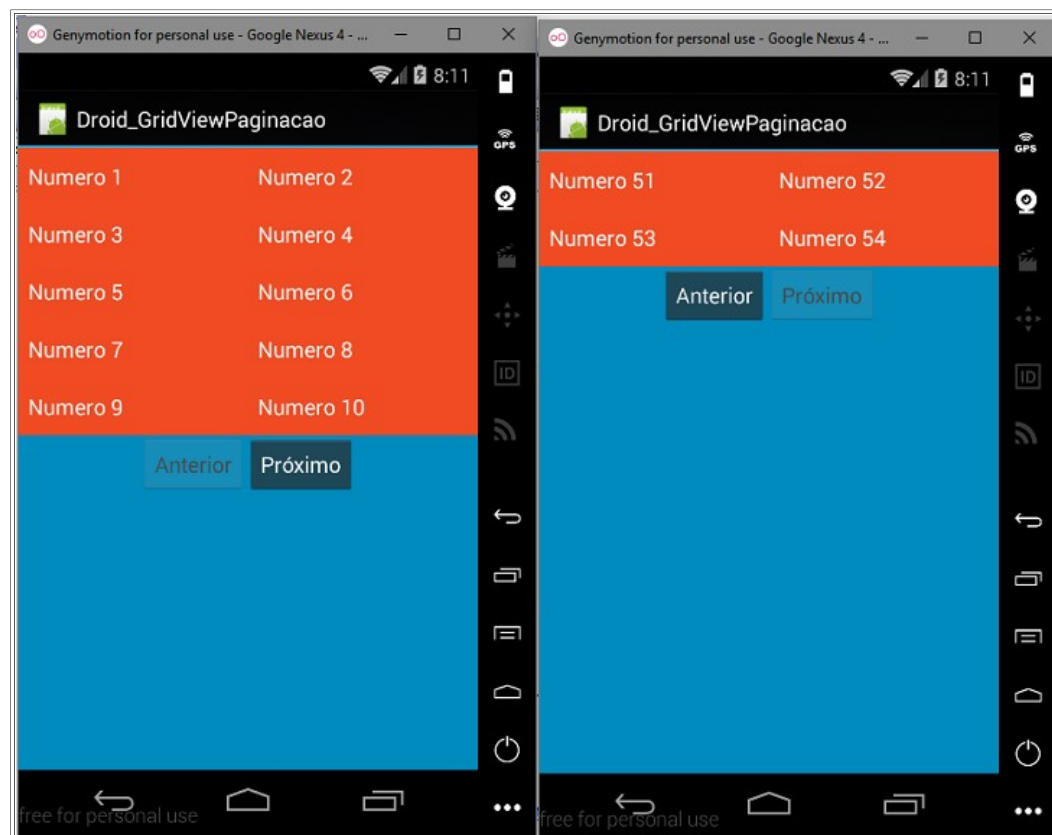
            ArrayList paginaDeDados = new ArrayList();

            if (paginaAtual == ULTIMA_PAGINA && ITEMS_RESTANTES > 0)
            {
                for (int i = itemInicial; i < itemInicial + ITEMS_RESTANTES; i++)
                {
                    paginaDeDados.Add("Numero " + i);
                }
            }
            else
            {
                for (int i = itemInicial; i < itemInicial + numDados; i++)
                {
                    paginaDeDados.Add("Numero " + i);
                }
            }
            return paginaDeDados;
        }
    }
}
```

Executando o projeto usando o emulador **Genymotion** iremos obter o seguinte resultado:

Na primeira imagem temos a primeira página e somente o botão **Próximo** ativo;

Após navegar até a última página vemos somente o botão **Anterior** ativo:



Pegue o projeto aqui : [Droid_GridViewPaginacao.zip](#) (sem as referências)

Considerai os corvos, que nem semeiam, nem segam, nem têm despensa nem celeiro, e Deus os alimenta; quanto mais valeis vós do que as aves? [Lucas 12:24](#)

[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](#)
- [VB.NET 2005 - Controles - Macoratti.net](#)
- [Seção de Jogos do site Macoratti .net](#)
- [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](#)
- <https://developer.android.com/reference/android/app/Activity.html>
- <https://developer.xamarin.com/api/type/Android.Widget.ProgressBar/>
- [Xamarin Android - ListView - Customizando um Adapter - Macoratti](#)
- [Xamarin Android - Usando SearchView com ListView - Macoratti](#)
- [Xamarin Android - ListView - Usando uma ListActivity - Macoratti](#)

[José Carlos Macoratti](#)