

# App BatLanterna

App feito em Xamarin que acessa os recursos de bateria e lanterna

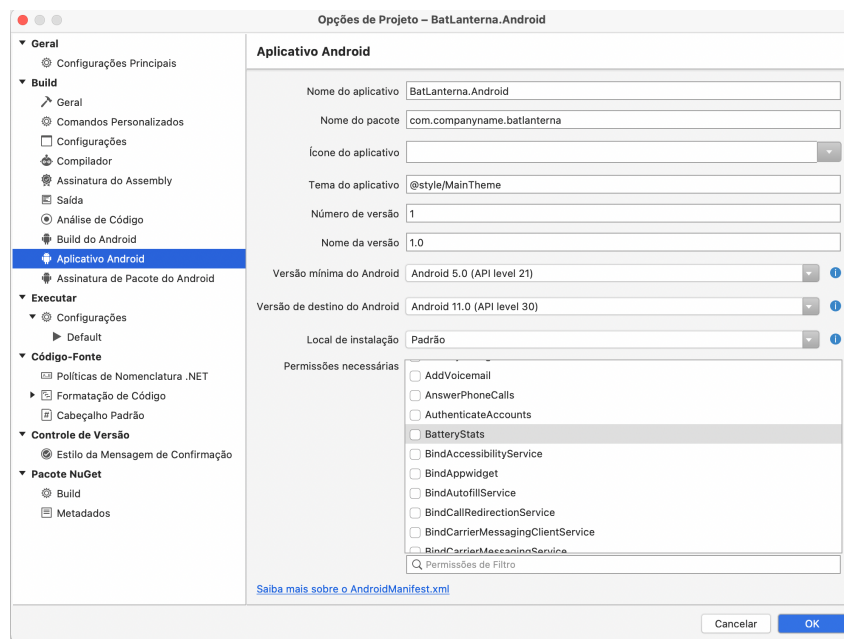
## Criando o projeto:

- Crie o projeto BatLanterna

## Ativando permissões no Android:

As permissões para acessar bateria, câmera, flash e vibração são necessárias e devem ser configuradas no projeto do Android. Para isso faça:

- Clique com o botão direito do mouse no projeto do Android e abra as propriedades do projeto. Em Manifesto do Android, localize a área Permissões necessárias: e marque as permissões indicadas abaixo. Isso atualizará automaticamente o arquivo AndroidManifest.xml.
  - BatteryStats
  - Camera
  - Flashlight
  - Vibrate



Obs: O iOS não exige mais essas configurações.

## Instalando pacotes:

- No Gerenciador de Soluções, selecione o projeto BatLanterna, clique com o botão direito do mouse e selecione Gerenciar Pacotes do NuGet...:
- Procure e instale os seguintes pacotes:
  - Xamarin.Essentials
  - Xam.Plugin.Battery

## Criando os estilos para os componentes visuais:

- Abra o arquivo App.xaml e digite o seguinte trecho em Application.Resources:

<Application.Resources>

```
<Style x:Key="lbl" TargetType="Label">
  <Setter Property="HorizontalOptions" Value="Center"/>
  <Setter Property="VerticalOptions" Value="Center"/>
  <Setter Property="HorizontalTextAlignment" Value="Center"/>
</Style>
```

```
<Style x:Key="lbl_titulo" TargetType="Label" BasedOn="{StaticResource lbl}">
  <Setter Property="FontSize" Value="40"/>
  <Setter Property="TextColor" Value="Orange"/>
</Style>
```

```
<Style x:Key="lbl_textoClaro" TargetType="Label" BasedOn="{StaticResource lbl}">
  <Setter Property="FontSize" Value="24"/>
  <Setter Property="TextColor" Value="White"/>
</Style>
```

```
<Style x:Key="lbl_textoColorido" TargetType="Label" BasedOn="{StaticResource lbl}">
  <Setter Property="FontSize" Value="24"/>
  <Setter Property="TextColor" Value="Orange"/>
</Style>
```

```
<Style x:Key="lbl_textoCinza" TargetType="Label" BasedOn="{StaticResource lbl}">
  <Setter Property="FontSize" Value="20"/>
  <Setter Property="TextColor" Value="Silver"/>
</Style>
```

</Application.Resources>

## MainPage.xaml:

- Abra o arquivo MainPage.xaml e digite os trechos em azul:

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="BatLanterna.MainPage"
    BackgroundColor="#232323">

    <StackLayout>
        <BoxView BackgroundColor="Silver" HeightRequest="60"/>
        <Frame BackgroundColor="Black" Padding="20,20,20,20" CornerRadius="0">
            <Label Text="BatLanterna"
                Style="{StaticResource lbl_titulo}"

            />
        </Frame>

        <Grid RowDefinitions="*,*,*,*,*,*,*"
            ColumnDefinitions="*,*"
            Margin="10"
            Padding="10">

            <Label
                Grid.Row="0" Grid.Column="0"
                Text="Lanterna"
                Style="{StaticResource lbl_textoCinza}"

            />

            <Switch
                x:Name="ligaLanterna"
                Grid.Row="0" Grid.Column="1"
                VerticalOptions="Center"
                HorizontalOptions="Center"
                OnColor="Green" ThumbColor="White"
                Toggled="ligaLanterna_Toggled"
            />

            <Label
                Grid.Row="1" Grid.ColumnSpan="2"
                Text="Situação da Bateria"
                Style="{StaticResource lbl_textoColorido}"
                Margin="10,20,10,10"
            />

            <Label
```

```
Grid.Row="2" Grid.Column="0"
Text="Carga Restante:"
Style="{StaticResource lbl_textoCinza}"
/>
```

```
<Label
  x:Name="lblCargaRestante"
  Grid.Row="2" Grid.Column="1"
  Text="..."
  Style="{StaticResource lbl_textoClaro}"
/>
```

```
<Label
  Grid.Row="3" Grid.Column="0"
  Text="Status Bateria:"
  Style="{StaticResource lbl_textoCinza}"
/>
```

```
<Label
  x:Name="lblStatus"
  Grid.Row="3" Grid.Column="1"
  Text="..."
  Style="{StaticResource lbl_textoClaro}"
/>
```

```
<Label
  Grid.Row="4" Grid.Column="0"
  Text="Fonte da Bateria:"
  Style="{StaticResource lbl_textoCinza}"
/>
```

```
<Label
  x:Name="lblFonte"
  Grid.Row="4" Grid.Column="1"
  Text="..."
  Style="{StaticResource lbl_textoClaro}"
/>
```

```
<Label
  x:Name="lblBateriaFraca"
  Grid.Row="5" Grid.ColumnSpan="2"
  Style="{StaticResource lbl_textoClaro}"

  FontAttributes="Italic"
  TextColor="Red"
/>
```

**</Grid**

**</StackLayout>**

**</ContentPage>**

### MainPage.xaml.cs:

- Abra o arquivo MainPage.xaml.cs e digite os trechos em azul:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;
```

```
using Plugin.Battery;
using Xamarin.Essentials;
```

```
namespace BatLanterna
{
    public partial class MainPage : ContentPage
    {

        public MainPage()
        {
            InitializeComponent();
            Carrega_Info_Bateria();
        }

        private async void Mudanca_Status_Bateria(object sender,
Plugin.Battery.Abstractions.BatteryChangedEventArgs e)
        {
            try
            {
                lblCargaRestante.Text = e.RemainingChargePercent.ToString() + "%";

                //Caso bateria esteja fraca, mostrar mensagem para usuário.
                if (e.IsLow)
                {
                    lblBateriaFrac.Text = "Atenção! A bateria está fraca!";
                }
                else
                {
                    lblBateriaFrac.Text = "";
                }

                switch (e.Status)
                {
```

```

        case Plugin.Battery.Abstractions.BatteryStatus.Charging:
            lblStatus.Text = "Carregando";
            break;

        case Plugin.Battery.Abstractions.BatteryStatus.Discharging:
            lblStatus.Text = "Consumindo bateria";
            break;

        case Plugin.Battery.Abstractions.BatteryStatus.Full:
            lblStatus.Text = "Carregada";
            break;

        case Plugin.Battery.Abstractions.BatteryStatus.NotCharging:
            lblStatus.Text = "Sem carregar";
            break;

        case Plugin.Battery.Abstractions.BatteryStatus.Unknown:
            lblStatus.Text = "Desconhecido";
            break;
    }

    switch (e.PowerSource)
    {
        case Plugin.Battery.Abstractions.PowerSource.Ac:
            lblFonte.Text = "Carregado";
            break;

        case Plugin.Battery.Abstractions.PowerSource.Battery:
            lblFonte.Text = "Bateria";
            break;

        case Plugin.Battery.Abstractions.PowerSource.Usb:
            lblFonte.Text = "USB";
            break;

        case Plugin.Battery.Abstractions.PowerSource.Wireless:
            lblFonte.Text = "Sem Fio";
            break;

        case Plugin.Battery.Abstractions.PowerSource.Other:
            lblFonte.Text = "Desconhecida";
            break;
    }
}

catch (Exception ex)

```

```

    {
        await DisplayAlert("Ocoreu um erro: \n", ex.Message, "OK");
    }
}

private async void Carrega_Info_Bateria()
{
    try
    {
        //Verificar se plugin é suporta pelo dispositivo.
        if (CrossBattery.IsSupported)
        {
            CrossBattery.Current.BatteryChanged -= Mudanca_Status_Bateria;
            CrossBattery.Current.BatteryChanged += Mudanca_Status_Bateria;
        }
        else
        {
            lblBateriaFrac.Text = "As informações sobre a bateria não estão
disponíveis.";
        }
    }
    catch (Exception ex)
    {
        await DisplayAlert("Ocoreu um erro: \n", ex.Message, "OK");
    }
}

private async void lanterna_onoff(object sender, ToggledEventArgs e)
{
    try
    {
        if (switchLanterna.IsToggled)
        {
            Vibration.Vibrate(TimeSpan.FromMilliseconds(250)); //Alerta 250
milisegundos.

            await Flashlight.TurnOnAsync();
        }
        else
        {
            Vibration.Vibrate(TimeSpan.FromMilliseconds(300)); //Alerta 300
milisegundos.

```



```
        await Flashlight.TurnOffAsync();
    }
}
catch (Exception ex)
{
    await DisplayAlert("Ocoreu um erro: \n", ex.Message, "OK");
}
}
}
```