

Dando a opção de cancelar ao usuário

Transcrição

Se vamos oferecer ao usuário a possibilidade de cancelamento da operação, precisamos que isto seja feito de forma a haver interação do usuário com a aplicação. Uma forma muito comum de fazer isto é colocar um botão de "Cancelar", e faremos isto ao lado do botão "Fazer Processamento" já existente.

Alteraremos o arquivo `MainWindow.xaml` do WPF, responsável pela interface gráfica. Vamos fazer uma pequena alteração, substituindo o botão "Fazer Processamento" por uma *grid*, um contêiner que permite o uso de linhas e colunas. Criaremos duas colunas, ambas com mesma largura, utilizando " * ", que define que todos com esta atribuição terão esta proporção.

Feito isto, vamos retirar o `DockPanel.Dock` do botão, pois ele passa a pertencer à `Grid`. Subiremos os códigos referentes ao botão para dentro da tag `<Grid>` que acabamos de criar. Acrescentaremos o botão de cancelar chamado `BtnCancelar`, repetindo-se os estilos de `Padding`, com `Click` como novo manipulador (`BtnCancelar_Click`). O texto que aparecerá será "Cancelar".

```
<Grid DockPanel.Dock="Bottom">
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="*" />
    <ColumnDefinition Width="*" />
  </Grid.ColumnDefinitions>

  <Button Name="BtnProcessar"
    Padding="10"
    Click="BtnProcessar_Click">
    Fazer Processamento
  </Button>

  <Button Name="BtnCancelar"
    Padding="10"
    Grid.Column="2"
    IsEnabled="False"
    Click="BtnCancelar_Click">
    Cancelar
  </Button>

</Grid>
```

Também definimos que este novo botão ocupará a segunda coluna do `Grid`. Quando abrimos a aplicação, ela não começa executando um processamento, portanto seu valor será `false`. Em relação ao manipulador de evento `BtnCancelar_Click`, podemos implementar algo no código do arquivo `MainWindow.xaml.cs`. Vamos recortá-lo do fim do código e mudar sua localização para perto do botão de processamento, apenas por questão de organização. Seu código fica assim:

```
private void BtnCancelar_Click(object sender, RoutedEventArgs e)
{
    BtnCancelar.IsEnabled = false;
}
```

Assim que o usuário clica no botão de cancelamento, inicialmente o botão deixa de ficar *enabled*, então, quando ele está disponível, a partir do clique pelo usuário, ele passará a ficar *disabled*. Quando ele fica disponível? Quando o processamento ocorre, ou seja, quando o usuário clica no botão "Fazer Processamento". Na parte que se refere ao início do processamento, portanto, incluiremos:

```
var inicio = DateTime.Now;  
  
BtnCancelar.IsEnabled = true;
```

Ainda não colocamos o `CancellationToken` nem o `CancellationTokenSource`, apenas alteramos a *view* para deixar a interface mais amigável para o usuário. Vamos verificar isto clicando em "Start" e rodando a aplicação, e continuar no próximo vídeo!