

Perfil

Transcrição

[00:00] Tudo bem com vocês? Eu sou Marcelo Oliveira, bem vindos de volta ao nosso curso de Xamarin com Visual Studio e parece que o pessoal da Aluracar gostou bastante do aplicativo que fizemos e eles pediram um novo requisito do aplicativo que é que o usuário vai poder tirar uma foto com o aparelho e colocar essa foto no próprio perfil do aplicativo e essa tarefa vai ser dividida em duas tarefas menores, primeiro tirar uma foto e em seguida exibir essa foto na tela.

[00:35] Vamos começar pela tarefa mais simples, vamos primeiro exibir uma foto, porque essa tarefa não exige que conheçamos recursos nativos do aparelho, do sistema operacional que no nosso caso é Android, como inicialmente não vamos ter a foto do usuário, vamos exibir uma imagem padrão que vai representar o perfil dele enquanto ele não tiver uma foto e que imagem é essa?

[01:03] Nós recebemos, a partir da Aluracar nós recebemos essa imagem aqui que vai ser exibida como imagem padrão do usuário, enquanto ele não tiver uma imagem definida, uma foto definida no perfil, o que vamos fazer?

[01:22] Adicionar no nosso projeto essa imagem, vamos ter que coloca-la em alguma pasta, que pasta é essa? Como estamos trabalhando com Android e já trabalhamos com imagem antes para fazer a imagem do logotipo da Aluracar vamos colocar essa imagem exatamente naquele local, que local é esse?

[01:46] Vamos abrir aqui o projeto “Droid”, vamos procurar em “Resources”, que é em inglês significa recursos, vamos procurar aqui “drawable” que temos duas imagens, que é o “aluracar” e o “icon.png”, vamos colocar exatamente aqui a nossa imagem do perfil, enquanto o usuário não tem uma foto definida.

[02:13] Eu vou clicar em “drawable”, vou adicionar novo item e vou procurar a pasta onde tem a minha imagem de perfil, seleciona a imagem, está adicionando, ele entrou aqui como Android, “Resources”, está pronto para ser utilizado na aplicação do Xamarin Forms. Como fazemos para exibir esta imagem no perfil do usuário?

[02:43] Vamos em “Views” que criamos para exibir o perfil, na “MasterView” temos um StackLayout, aqui em cima e definimos aqui uns Labels, botões, vamos adicionar em cima, como o primeiro item do StackLayout uma imagem, como é a imagem em inglês? É “image”, então vamos adicionar o elemento “image” e definir a fonte, a origem dessa imagem, o arquivo de origem da imagem que vai ser a propriedade “source” e eu coloco o nome do arquivo que vai ser “perfil.png”.

[03:36] Como eu estou utilizando o “perfil.png”, como eu o coloquei na pasta padrão do Android eu não preciso definir o caminho dessa imagem, posso colocar simplesmente o nome do arquivo que ele vai encontrar.

[03:51] Vamos rodar a aplicação e vamos ver como que isso é exibido na tela, eu vou inserir o usuário “joao@alura.com”, a senha “alura123”, vou entrar e vou clicar no perfil. No perfil apareceu a imagem que é o perfil, que é a imagem de perfil do usuário enquanto ele não tirou uma foto ainda para colocar.

[04:25] Vamos colocar também essa imagem na segunda página, no StackLayout vamos colocar um “image” também, o “Source” vai ser “perfil.png”. Agora rodamos a aplicação de novo. Agora eu insiro o usuário “joao@alura.com.br”, senha “alura123”, vou entrar, clicar no perfil. Vou para edição do perfil e apareceu em cima a imagem do usuário.

[05:10] Não é uma boa ideia deixar a imagem “perfil.png” fixa no nosso código “xaml”, por quê? Porque depois que o usuário tirar uma foto, ela vai ser exibida nesse local. Então essa fonte, essa origem de imagem, esse “Source” não pode

ficar fixo, eu tenho que utilizar uma coisa que pode ser trocada pela imagem que vai vir da câmera depois. Então o que eu vou fazer?

[05:42] Eu vou ter que criar um Binding para uma propriedade que vai ficar no ViewModel que vai ser trocada depois, quando conseguirmos finalmente fazer a captura de imagem pela câmera do dispositivo, o que vamos fazer?

[06:00] Eu vou criar um “Binding” e definir uma nova propriedade que vou ter que criar no ViewModel. Vou chamar essa propriedade “FotoPerfil”. Isso eu vou ter que colocar na outra página também. Então eu troco também no lugar desse “perfil.png”, eu coloco um “Binding FotoPerfil”.

[06:29] Agora “FotoPerfil” vai ser amarrado, vai ter o Binding para a propriedade “Source”. O que é o Source? Será que o Source é uma string? Será que essa propriedade da classe “Image” é um é uma string? Vamos ver.

[06:45] Eu vou deixar o mouse em cima e vou clicar com “F12” para irmos para as definições dessa propriedades na classe de imagem. Clicando agora no “F12”, o Source é um “ImageSource”, que é um tipo que está dentro do “Xamarin.Forms”, eu tenho que definir no meu ViewModel uma propriedade que é do tipo, que é da classe “ImageSource”. Então vamos lá no ViewModel e vamos definir essa nova propriedade.

[07:15] Aqui em cima, vamos ver, eu vou criar uma propriedade chamada “FotoPerfil”, vou criar um “ImageSource” e aqui “FotoPerfil”, aqui “FotoPerfil” de novo só que maiúsculo e agora eu vou ter que definir um valor padrão, um valor default. Eu vou colocar aqui a string “perfil.png”.

[07:59] Para não deixar essa propriedade muito aberta vamos deixar aqui o “setter”, esse assessor “set”, para essa propriedade vamos deixar ele como privado. Então “private set”. Vamos rodar para ver se não quebra nada, eu coloco o usuário “joao@alura.com.br”, “alura123”. Funcionou, não quebrou nada, não parou de funcionar, agora temos a exibição do perfil a partir de uma propriedade que está lá no nosso ViewModel.