

Intent

Transcrição

[00:00] Agora temos que falar para câmera onde gravar a imagem que ela capturar do nosso usuário. O que eu preciso fazer? Eu tenho que utilizar o “intent” como o mensageiro entre a nossa aplicação e a atividade da aplicação externa, que no caso é a câmera. O nosso “intent” vai levar essa informação, esse dado de que temos que gravar esse arquivo, essa imagem do usuário da câmera e um determinado arquivo, vamos fazer isso colocando dados dentro de um “intent”, como é isso?

[00:41] Eu pego o “intent” e a partir disso eu coloco dados nele, como eu coloco dados? Eu utilizo o comando “put” que é colocar em inglês e coloco “PutExtra”, que significa colocar dados extra, colocar alguma coisa a mais, alguma informação a mais para câmera poder fazer o trabalho de gravar a imagem em arquivo e aqui dentro do “PutExtra” eu tenho que falar qual é o tipo de informação que eu tô passando. Aqui eu tenho como parâmetro “string” e o valor, o nome do parâmetro, o nome do dado que eu estou passando e o valor.

[01:26] Como eu não sou muito bom de guardar nomes, eu fico muito feliz quando aparece aqui a opção de utilizar uma constante e felizmente para gente temos esse “MediaStore” que é uma classe que tem uma série de constantes, eu vou utilizar essa classe “MediaStore” no método “PutExtra” também, eu faço “MediaStore.” e eu procuro o quê?

[01:53] O “ExtraOutput” que vai dizer para o sistema Android que eu tô passando uma intenção, um “intent” com dados extra e esse dado é do tipo “ExtraOutput”, ou seja, é um local de saída. O Output é uma saída de dados, é onde vai ser guardado o resultado final da captura da foto.

[02:24] Em seguida vou passar a informação do arquivo, do local onde gravar essa informação. O segundo parâmetro é uma URI, só que ela não é uma URI qualquer, ela é um URI do Android, não estamos no sistema operacional do Windows, estamos trabalhando com Android então eu tenho que tem que acessar a classe de URI do Android. O Xamarin converte qualquer tipo do Android no código CSharp para trabalharmos. Como acessamos a classe de URI do Android?

[03:01] Eu faço assim, “Android.Net” e aqui “Uri” e a partir do Uri eu tenho um método chamado “FromFile” que vai converter um tipo que é arquivo, que é file para Uri, quando eu faço “FromFile” eu tenho aqui um parâmetro que é um File, que é um arquivo, só que ele é um “Java.IO”.

[03:30] De novo estamos vendo o tempo todo nessas classes que foram portadas do Java, do Android, para o CSharp, para trabalhar dentro do Xamarin, que porta, desde o começo ele já portou todas essas classes para podermos trabalhar com o sistema operacional do Android.

[03:58] Aqui dentro eu tenho que colocar o quê? Eu tenho que colocar o arquivo de imagem. Eu não tenho esse arquivo de imagem ainda, mas eu vou definir aqui uma variável, vou colocar “arquivoImagem” e declarar ela aqui em cima, antes do “PutExtra” eu vou declarar aqui um “Java.IO.File” e aqui “arquivoImagem”. Esse “arquivoImagem” vai ser um novo “Java.IO.File” e aqui dentro eu coloco o quê?

[04:37] Eu coloco a string do nome do arquivo, que eu vou chamar de “MinhaFoto.jpg”, mas reparem que eu não posso sair gravando um arquivo em qualquer lugar do Android, sem mencionar qual é a pasta, qual é o diretório, eu tenho que colocar aqui no nome do arquivo, antes do arquivo tem que mencionar também qual é o diretório, eu vou fazer isso agora, eu coloco aqui como primeiro parâmetro o “diretorio” que vai ser uma variável que eu vou declarar agora, eu declaro aqui em cima do diretório, que vai ser o quê?

[05:18] Ele vai ser um “Java.IO.File” também. Essa classe “File” funciona tanto para arquivo como para diretório, tanto que quando declaramos o diretório eu posso fazer “diretório.isDirectory”, eu tenho “isDirectory” e tenho “isFile”, essa informação vai dizer se o meu objeto é um arquivo ou diretório.

[05:51] Agora eu vou declarar o diretório como um “newJava.IO.File(“ e aqui eu vou passar o quê? Eu vou passar qual é a pasta base, a pasta raiz onde eu vou gravar esse diretório, como eu não quero colocar aqui a string desse caminho do diretório na mão, felizmente trabalhamos, podemos trabalhar com aquelas constantes do Android, o que eu faço?

[06:18] Eu coloco aqui no “Android.OS.Environment” e aqui eu vou pegar qual é a pasta pública, onde é a pasta que eu posso realmente gravar os dados, porque como o sistema operacional é fechado, principalmente o sistema operacional do smartphone é muito fechado para as aplicações, eu tenho que saber exatamente onde eu posso guardar, onde eu posso gravar os dados que a minha aplicação está utilizando.

[06:52] Então eu vou fazer aqui, eu vou pegar um método chamado “getExternalStoragePublicDirectory”, aqui ele vai me dar o diretório onde eu posso gravar os dados e esse nome aqui “getExternalStoragePublicDirectory “ é um método e exige também que eu passe qual é a subpasta, porque ele tem uma série de pastas, como ele é um diretório público eu não posso simplesmente jogar os dados lá em qualquer lugar, eu quero guardar numa pasta específica para imagens e que pasta é essa?

[07:25] É a pasta chamada “Android.OS.Environment.” e eu tenho acesso a uma constante chamada “DirectlyPictures”, percebam que temos várias opções, documentos, filmes, notificações, podcast, eu vou selecionar a pasta específica para imagens.

[07:53] Passamos também um segundo parâmetro que é uma pasta local, que é onde vão ficar gravadas as imagens, ou melhor, a imagem do perfil do usuário, eu coloco aqui “Imagens” e faltou fechar aqui também, muito bem, agora temos aqui o diretório declarado, definimos aqui o diretório e esse diretório é utilizado depois aqui embaixo para ser a base onde a imagem do perfil do usuário, que é a “MinhaFoto.jpg”, onde essa imagem vai ser gravada.

[08:32] Agora tem um problema, esse diretório estamos falando para ele guardar em um subdiretório chamado “Imagens”, só que na primeira vez que esse código rodar esse diretório ainda não vai existir, o que eu preciso saber é se esse diretório existir eu quero que ele utilize, mas se não existir ainda eu quero que ele crie, como eu faço isso?

[08:59] Primeiro o que eu faço é verificar a existência dele, “diretorio.Exists” que vai me retornar uma “boolean”, vai retornar um verdadeiro ou falso, se ele não existir “if(diretorio.Exists())” eu quero criar esse diretório no Android, como que eu faço para criar?

[09:22] Eu pego “diretório.mkdir”, o “mkdir” vai criar aquela pasta, a pasta assumindo se verificarmos que ela não existe, agora existe outro método chamado “mkdir” que vai criar toda a estrutura acima se ela não existir, nós garantimos que todos os pais, os avós, toda a estrutura daquele diretório vai ser criado se não existir, com isso temos aqui todo o código necessário para passar para o nosso “intent” qual é o local de gravação da imagem do perfil do usuário.

[10:10] Agora olhando aqui nesse método, o “TirarFoto” estamos vendo que ele está muito grande, precisamos dar uma melhoria nele, uma refaturada nele para poder deixar ele mais limpo, como fazemos isso?

[10:25] Selecionamos o código que é relativo a manipulação de arquivos a partir do diretório até aqui embaixo “arquivoImagem” e vamos extrair o método, só que antes disso eu quero declarar fora desse bloco, eu quero declarar aqui o “arquivoImagem”, eu declaro aqui “arquivoImagem”, removo aqui a declaração do “arquivoImagem” que está em cima, que está aqui embaixo aliás e eu seleciono aquele bloco de código, com o direito “Quick Actions” e eu vou extrair o método, eu vou chamar esse método de “PegarArquivoImagem”.

[11:08] O nosso código agora ficou um pouco mais limpo, agora será que a nossa aplicação já funciona? Vamos testar, rodando agora, legal, agora vou passar aqui o usuário “joao@alura.com.br”, senha “alura123” vou entrar agora, vou lá no perfil e vou verificar a câmera, vamos lá, tirar foto, apareceu aquela imagem, vou tirar a foto e vou escolher “Ok”, quando eu faço isso, ele voltou para a aplicação, mas ainda não substituiu a nossa foto, o que eu preciso fazer?

[11:47] Estamos chamando a câmera, mas eu verifiquei que não estamos tratando o retorno da nossa atividade, não estamos sabendo que fazer na volta, temos que tratar o retorno, o resultado da atividade do aplicativo de câmera e como fazemos isso?

[12:09] Vamos entrar aqui na classe MainActivity, vamos fazer um “Override”, eu faço um “Override” em “ActivityResult”, aqui em baixo, muito bem, agora o que vamos fazer? Vamos ter que acessar aquela variável de “arquivoImagem”, só que ele está declarado aqui dentro do bloco, desse método pegar “arquivoImagem”, ou melhor, aqui no tirar foto, está declarado em “TirarFoto”, temos que fazer o quê?

[12:47] Mover ele para o escopo acima, vamos mover o “arquivoImagem”, vamos tirar do escopo do método e colocar no escopo da classe, eu vou declarar também como estático porque não vai ser a mesma instância que vai chamar esse método, esse evento “OnActivityResult”, eu tenho que declarar ele como estático, “static Java.IO.File arquivoImagem”, muito bem, agora eu vou consegui acessar o “arquivoImagem” a partir do “ActivityResult”.

[13:23] Agora que eu tenho o “arquivoImagem” podendo ser acessado aqui no resultado da atividade, o que eu preciso fazer é passar essa imagem lá para o outro projeto, para o projeto PCL aqui em cima, “TestDrive (Portable)”, que é onde tudo começou, onde eu cliquei no botão para tirar foto, eu preciso passar essa imagem lá para cima, como eu faço essa comunicação entre esses dois projetos?

[13:50] Podemos utilizar o Messaging Center, o “MessagingCenter” até agora foi o nosso método favorito para trocar mensagens, eu quero que ele envie uma mensagem nova com o arquivo de imagem, que arquivo que é esse? É um arquivo do tipo “Java.IO.File” e eu tenho que passar o quê?

[14:17] Eu tenho que passar qual é a instância que tem esse dado e a instância é “arquivoImagem”, eu vou passar umas String que vai ser o identificador desse tipo de mensagem, que eu vou chamar de “TirarFoto”.

[14:34] Agora coloca um breakpoint nessa linha, aqui também, vou rodar a aplicação e ver se ele retorna e passa por aqui para podermos fazer esse envio da mensagem do arquivo, vou colocar o usuário “joao@alura.com.br”, senha “alura123”, vou entrar agora, vou lá no perfil, clicar aqui no perfil e tirar foto, muito bem, tirando a foto e vou dar um “OK”.

[15:10] Agora ele passou aqui para o método “OnActivityResult”, estamos conseguindo obter o resultado de volta e quando eu aperto “F5” ele passa por essa linha e chega no “arquivoImagem”, que foi criado aqui, foi instanciado, agora vamos ver como pegar esse arquivo de imagem, pegar essa imagem e tratar ele lá no outro projeto “TestDrive (Portable)”, para poder exibir a imagem no perfil do usuário.