

Android

Galeria de Fotos e Câmera

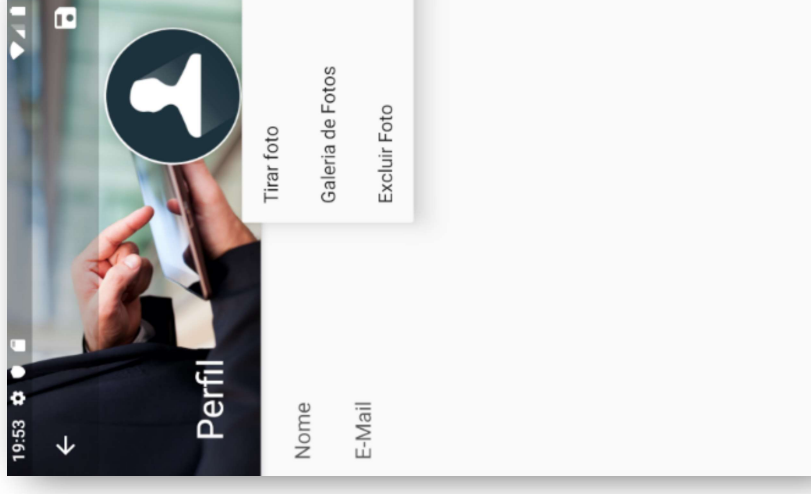


Adicionando Fotos

A tela de Perfil permite adicionar Fotografias que podem vir da Galeria de Fotos ou serem obtidas da Câmera do celular.

Para implementar esta funcionalidade em uma aplicação, deve ser determinado onde serão armazenadas as fotografias e como referencia-las.

Em nosso exemplo utilizaremos o SQLite para o armazenamento e recuperação das fotografias associadas aos registros armazenados pela aplicação.



Adicionando Fotos

SENAI-SP

Para facilitar a obtenção da Fotografia e o recorte da área de interesse dentro da foto será utilizada uma API chamada **ImagePicker**.

Para isto será necessário configurar o arquivo **build.gradle** conforme abaixo:

```
dependencies {  
  
    // Implementa Image Capture/Crop  
    implementation 'com.linchaolong.android:imagepicker:1.5'
```

No Fragmento que implementa a tela de Perfil inicializaremos uma instância de **ImagePicker**.

```
private val imagePicker = ImagePicker()
```

```
com.linchaolong.android:imagepicker:ImagePicker  
@Contract(pure = true)  
public ImagePicker()  
  
Inferred annotations: @org.jetbrains.annotations.Contract(pure = true)  
Gradle: com.linchaolong.android:imagepicker:1.5@aar
```



EMPREENDA
RAPIDO

SEBRAE

SENAI

Adicionando Fotos

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="br.senai.sp.informatica.albumsmusicais">

    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/albums"
        android:label="@string/AlbumsMusicais"
        android:roundIcon="@drawable/albums"
        android:supportRtl="true"
        android:theme="@style/AppTheme"
        tools:ignore="AllowBackup, GoogleAppIndexingWarning"
        android:name=".Main">

        <activity android:name=".view.activity.MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

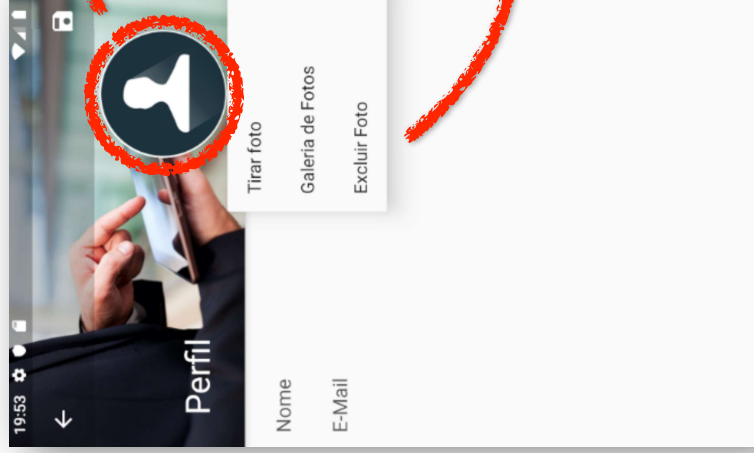
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name="com.linchaolong.android.imagepicker.cropper.CropImageActivity"
            android:theme="@style/Base.Theme.AppCompat" />
    </application>
</manifest>
```

Registra as permissões para o acesso à Câmera e para leitura de arquivos de imagem.

Registra a Activity que edita a fotografia permitindo o recorte da área de interesse dentro da fotografia selecionada.

Adicionando Fotos



```
ivFoto.setOnClickListener { it: View!
    val popUpMenu = PopUpMenu(context, it)
    popUpMenu.menuInflater.inflate(R.menu.menu_foto, popUpMenu.menu)
    popUpMenu.setOnMenuItemClickListener { item ->
        when (item.itemId) {
            R.id.ac_tirar_foto -> {
                imagePicker.setTitle("Tira Foto")
                imagePicker.setCropImage(true)
                imagePicker.startCamera(fragment: this@PerfilFragment, Camera())
            }
            R.id.ac_galeria_foto -> {
                imagePicker.setTitle("Escolha o Visualizador de Fotos")
                imagePicker.setCropImage(true)
                imagePicker.startGallery(fragment: this@PerfilFragment, Camera())
            }
            else -> {
                it.invalidate()
                (it as ImageView).setImageBitmap(null)
            }
        }
        true ^setOnMenuItemClickListener
    }
    popUpMenu.show()
}
```

O **PopUpMenu** é associado ao click na Foto

Nas opções do **PopUpMenu** estão associadas as chamadas à Galeria de Fotos e Câmera.

EMPREENDA
RAPIDO

SEBRAE

SENAI

Adicionando Fotos

SENAI-SP

Após a Câmera do celular capturar a fotografia, sua imagem será salva em um diretório temporário e em seguida o método **onActivityResult** será chamado para que a imagem possa ser editada pelo **ImagePicker**.

```
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {  
    | imagePicker.onActivityResult( fragment: this, requestCode, resultCode, data)  
}
```

EMPREENDA
RAPIDO

SEBRAE

SENAI

Adicionando Fotos

SENAI-SP

```
internal inner class Camera : ImagePicker.Callback() {  
    override fun onPickImage(imageUri: Uri) {}  
  
    override fun onCropImage(imageUri: Uri?) {  
        capdUri = imageUri  
        Glide.with( fragment: this@PerfilFragment)  
            .load(capdUri)  
            .fitCenter()  
            .circleCrop()  
            .into(ivFoto)  
    }  
  
    override fun cropConfig(builder: CropImage.ActivityBuilder) {  
        builder.setTitle("Recorte a Foto")  
            .setCropShape(CropImageView.CropShape.RECTANGLE)  
            .setAspectRatio( aspectRatioX: 5, aspectRatioY: 5)  
    }  
  
    override fun onPermissionDenied(requestCode: Int, permissions: Array<String>?,  
        grantResults: IntArray?) {  
        LongToast( text: "Permissão negada!")  
    }  
}
```

Esta classe implementa os mecanismos para a configuração de Activity de edição da Imagem e o método que recebe a foto recortada e pronta para o armazenamento.

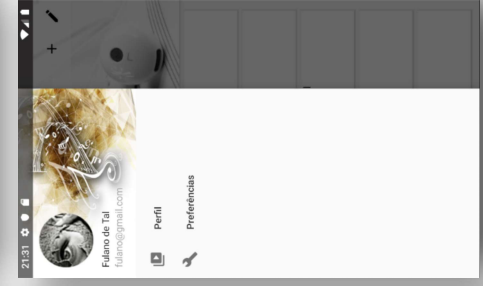
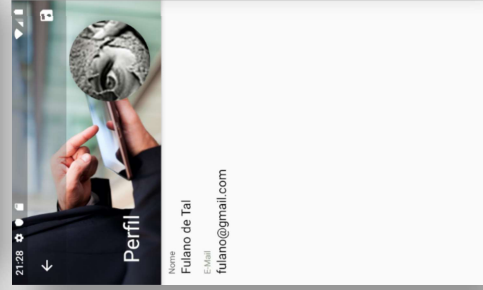
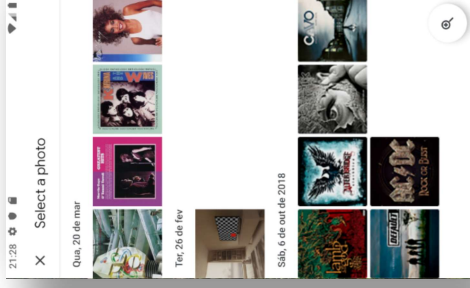
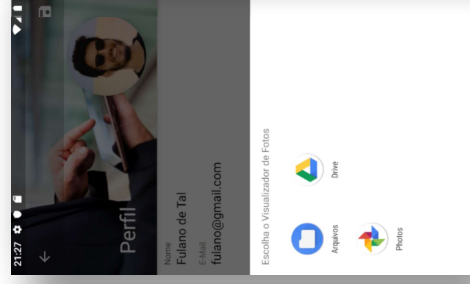
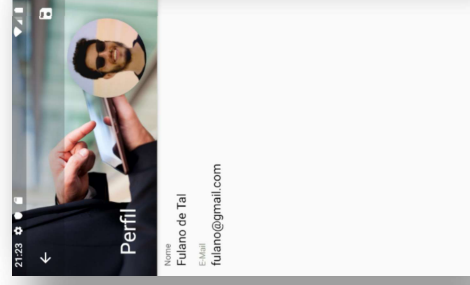
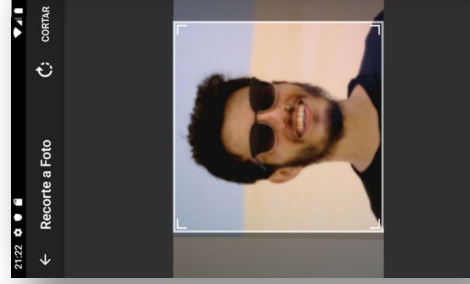
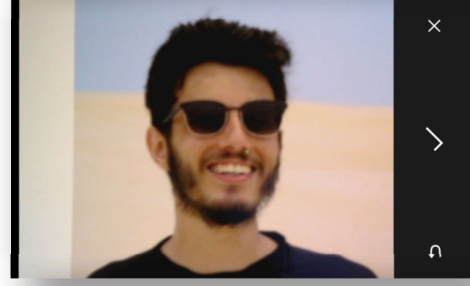
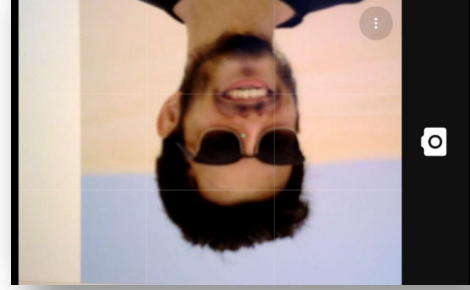
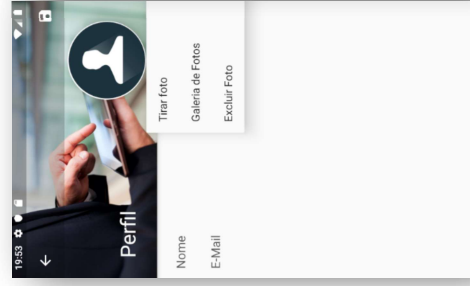
Este método trata a falha na solicitação de permissões para acesso à Câmera.

EMPREENDA
RAPIDO

SEBRAE

SENAI

Adicionando Fotos



EMPREENDA
RAPIDO

SEBRAE

SENAI

SÃO PAULO
GOVERNO DO ESTADO

Secretaria de
Desenvolvimento Econômico