

Multimídia

- Playing Audio
- Watching Video
- Recording Audio
- Using Camera

Referências

- DEITEL, Harvey, DEITEL, Paul, DEITEL, Abbey, MORGANO, Michael. Android para programadores: uma abordagem baseada em aplicativos. Porto Alegre: Bookman. 2013.
- Programming Handheld Systems – Prof. Adam Porter of University of Maryland (<https://www.coursera.org/course/android>)
- <http://developer.android.com>
- <https://developers.google.com/university/courses/mobile>
- <http://oreilly.com/training/androidapps>

Multimídia

- O Android oferece suporte para codificação e decodificação de uma variedade de formatos de mídia
- É possível reproduzir e gravar áudio, imagens estáticas e vídeos

Algumas *Multimedia Classes*

- AudioManager & SoundPool
- RingtoneManager & Ringtone
- MediaPlayer
- MediaRecorder
- Camera

AudioManager

- Carrega e executa efeitos de som
- Gerencia volume
- Gerencia periféricos

SoundPool

- Representa uma coleção de *streams* de áudio
- Permite mixar e tocar múltiplos sons simultaneamente

Ringtone e RingtoneManager

- *RingtoneManager* fornece acesso a clipes de áudio usados para receber chamadas telefônicas, notificações, alarmes, etc.
- Permite as aplicações obterem e executarem *ringtones*, bem como toca-los e interrompe-los

MediaPlayer

- Controla a reprodução de áudio e vídeo
- Permite que as aplicações controlem a reprodução em segundo plano (*playback*)
- Funciona de acordo com uma máquina de estado complexa:

`http://developer.android.com/
reference/android/media/MediaPlayer.html`

Alguns Métodos do *MediaPlayer*

- `setDataSource()`
- `prepare()`
- `start()`
- `pause()`
- `seekTo()`
- `stop()`
- `release()`

VideoView

- É uma *SurfaceView* para exibir arquivos de vídeo
- Pode carregar vídeo de várias fontes
- Fornece várias opções de exibição e funções de conveniência

MediaRecorder

- Usado para gravação de áudio e vídeo
- Funciona de acordo com uma máquina de estado:

`http://developer.android.com/
reference/android/media/MediaRecorder.html`

Alguns Métodos do *MediaRecorder*

- `setAudioSource()`
- `setVideoSource()`
- `setOutputFormat()`
- `prepare()`
- `start()`
- `stop()`
- `release()`

Camera

- Cliente para o serviço *Camera* que gerencia o hardware da câmera real
- Gerencia configurações de captura de imagem
- Inicia e interrompe a pré-visualização
- Tira fotos

Camera Permissions

```
<uses-permission  
    android:name="android.permission.CAMERA" />  
<uses-feature  
    android:name="android.hardware.camera" />  
<uses-feature android:name=  
    "android.hardware.camera.autofocus" />
```

Utilizando a Câmera

- Obter instância da classe *Camera*
- Definir os parâmetros necessários
- Ajustar a exibição da pré-visualização (*preview*)
- Iniciar a pré-visualização
- Tirar uma foto e processar os dados da imagem
- Liberar a câmera quando não estiver mais em uso