

OUTROS RECURSOS DO ANDROID



“Eu estou fazendo um sistema operacional livre (apenas um hobby, não vai ser grande e profissional como o gnu) para clones AT 386 (486)” (Linus Torvalds)

CONTEÚDO

- **LOGCAT**
- **GRIDVIEW E
WEBVIEW**
- **BROADCASTRECEIVER**



UTILITÁRIO LOGCAT

- Não existe `System.out.println()`, mas existe o LogCat
- LogCat permite escrever mensagens no Log do Android
- Níveis de severidade

Verbose	<code>Log.v(tag, message)</code>
Debug	<code>Log.d(tag, message)</code>
Info	<code>Log.i(tag, message)</code>
Warning	<code>Log.w(tag, message)</code>
Error	<code>Log.e(tag, message, exception)</code>



- Para ver no Eclipse:

– **Window > Show View > Other > Android > LogCat**

GRIDVIEW

- Exibe Views na forma de grade
- Todas as colunas têm a mesma largura
- Faz scrolling do conteúdo
- Atributos

Atributo	Descrição
android:columnWidth	Largura de cada coluna, em pixels
android:numColumns	Determina a quantidade de colunas (número). Use "auto_fit" para autoajuste.
android:verticalSpacing	Espaço vertical entre itens, em pixels
android:horizontalSpacing	Espaço horizontal entre itens, em pixels
android:stretchMode	Quando numColumns="auto_fit", indica como o Grid deve esticar para preencher o espaço do Grid. O valores permitidos são: none, columnWidth (expande a largura da coluna), spacingWidth (expande o espaçamento entre



- Exemplo:
 - <http://developer.android.com/resources/tutorials/views/hello-gridview.html>

WEBVIEW E WEBSETTINGS

- **WebView**
 - Apresenta uma página de web
 - Usa o mesmo componente que o navegador do celular
 - Necessário permissão `android.permission.INTERNET`
- **WebSettings**
 - Permite configurar o WebView
 - Permite JavaScript, Salvar senhas, etc.
- **Métodos principais**

Métodos WebView		Métodos WebSettings
WebSetting getSettings()	void loadUrl(String)	void setSavePassword(boolean)
boolean canGoBack()	String getTitle()	void setSaveFormData(boolean)
boolean canGoForward()	String getUrl()	void setJavaScriptEnabled(boolean)
boolean canZoomIn()	void goBack()	boolean getSavePassword()
boolean canZoomOut()	void goForward()	boolean getSaveFormData()
void clearFormData()	void zoomOut()	boolean getJavaScriptEnabled()
void clearHistory()	void zoomIn()	



WEBVIEW - EXERCÍCIO

- Criar um projeto que abra a página da UNOESC

- Dentro do método onCreate() da Activity

```
WebView web = new WebView(this);  
WebSettings webSettings = web.getSettings();  
webSettings.setSavePassword(false);  
webSettings.setSaveFormData(false);  
webSettings.setJavaScriptEnabled(true);  
webSettings.setSupportZoom(false);  
web.loadUrl("http://www.unoesc.edu.br");  
setContentView(web);
```

- Altere o AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<manifest ... >  
    <application ... >  
        <activity ... >  
            ...  
        </activity>  
    </application>  
    <uses-permission android:name="android.permission.INTERNET" />  
</manifest>
```



GALLERY E IMAGESWITCHER

- **Gallery**
 - Mostra lista de componentes com rolagem horizontal
 - Mantém o componente selecionado sempre no centro
- **ImageSwitcher**
 - Exibe imagens e cria efeitos ao alterná-las



BROADCASTRECEIVER

- Chamado pelo Android para reagir a mensagens broadcast
- Usado para executar tarefas rápidas em segundo plano
- Não interromper o usuário por meio de activities
- O que são mensagens broadcasts?
 - São mensagens enviadas pelo Android quando ocorre algum evento do sistema ou de uma aplicação
 - Interceptada pelos BroadcastReceiver's apropriados
 - Definido pelos <intent-filter>



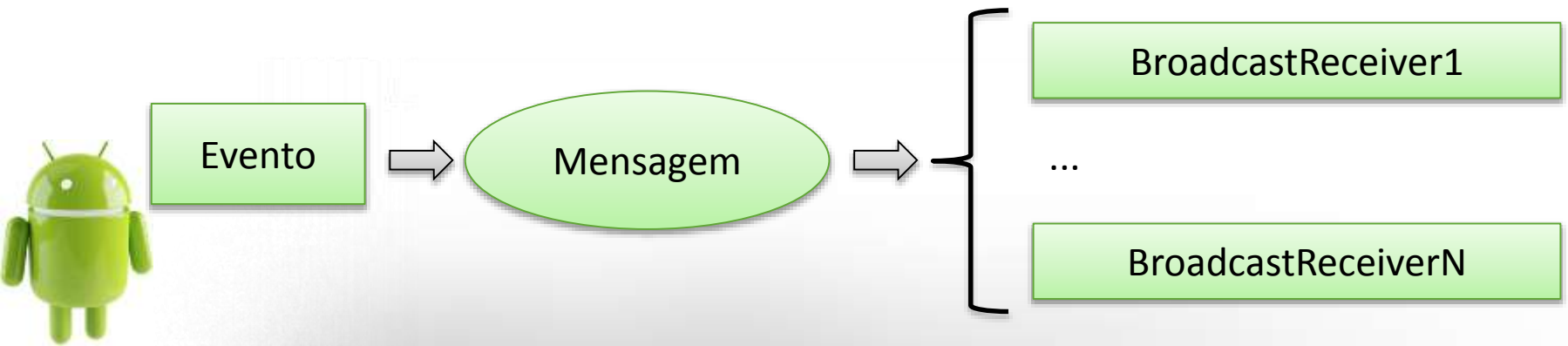
TIPOS DE BROADCAST

- **Ordered ou Síncrono**

- Os receivers são executados em sequência e a falha de um interrompe o processamento dos demais

- **Normal ou Assíncrono**

- Todos os receivers executam simultaneamente em paralelo e de forma independente
- Método para disparar: `Context.sendBroadcast(Intent)`



CONFIGURAÇÃO DO RECEIVER

- É necessário adicionar o elemento `<receiver>` dentro de `<application>` no `AndroidManifest.xml`

```
<receiver android:name="NomeReceiver">
    <intent-filter>
        <action android:name="com.example.broadcastreceiver.ACTION"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</receiver>
```

- Implementar uma classe filha de `BroadcastReceiver` com o callback `onReceive(Context, Intent)`

```
public class NomeReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        // ...
    }
}
```

- **Atenção: `onReceive` deve retornar em menos de 10 s(timeout).**

DISPARANDO MENSAGENS BROADCAST

- Criar uma Intent com a mensagem
- Enviar via **Context.sendBroadcast(Intent)**
 - Context é uma classe base de activity, logo o método pode ser chamado diretamente

```
Intent intent =  
    new  
        Intent("com.example.broadcaster.ACTION")  
    ;  
sendBroadcast(intent);
```



MENSAGENS DE BROADCAST DO SISTEMA

- Mensagens broadcast enviadas apenas pelo sistema
 - Não podem ser enviadas pelo método `sendBroadcast()`

android.intent.action	Condição de envio (pelo sistema)
WALLPAPER_CHANGED	Papel de parede alterado
BOOT_COMPLETED*	Sistema concluiu o bootstrap
ACTION_SHUTDOWN	Sistema está desligando (antes de desligar o aparelho)
BATTERY_CHANGED**	Estado de carga da bateria mudou
BATTERY_LOW	Estado de carga da bateria baixo
BATTERY_OKAY	Estado de carga da bateria voltou a níveis normais
ACTION_POWER_CONNECTED	Carregador conectado
ACTION_POWER_DISCONNECTED	Carregador desconectado

* Requer permissão `android.permission.RECEIVE_BOOT_COMPLETED`

** Não podem ser usados com `<intent-filter>`



MENSAGENS DE BROADCAST DO SISTEMA

<code>android.intent.action.*</code>	Condição de envio (pelo sistema)
<code>TIME_SET</code>	Data e/ou hora do sistema foram alterados
<code>TIMEZONE_CHANGED</code>	Fuso horário foi modificado
<code>TIME_TICK**</code>	Hora do sistema mudou normalmente (enviado a cada minuto)
<code>SCREEN_ON</code>	Tela do dispositivo acendeu (estava inativa, mas usuário ativou)
<code>SCREEN_OFF</code>	Tela do dispositivo apagou (normal para economizar bateria)
<code>PACKAGE_ADDED</code>	Uma nova aplicação foi instalada no dispositivo
<code>PACKAGE_REMOVED</code>	Uma aplicação foi removida do dispositivo
<code>PACKAGE_REPLACED</code>	Uma aplicação foi substituída. Normalmente por upgrade de versão.
<code>PACKAGE_RESTARTED</code>	Uma aplicação foi reiniciada, tendo seus processados terminados.
<code>PACKAGE_DATA_CLEARED</code>	Dados de uma aplicação foram apagados. Enviado após <code>PACKAGE_RESTARTED</code> .

**** Não podem ser usados com `<intent-filter>`**



CONFIGURANDO RECEIVERS VIA CÓDIGO

- Indicado quando um receiver deve executar apenas quando uma activity está executando
- É possível registrar dinamicamente via código
 - `registerReceiver(BroadcastReceiver, Intent)`
 - `unregisterReceiver(BroadcastReceiver)`
- Recomendável:
 - Registrar em `Activity.onCreate()` ou `onResume()`
 - Desregistrar no `Activity.onDestroy()` ou `onPause()`
- Note que são necessários para actions `TIME_TICK` e `BATTERY_CHANGED` da tabela

REFERÊNCIAS

- LECHETA, Ricardo R. **Google android:** aprenda a criar aplicações para dispositivos móveis com o android SDK. 2. ed., rev. e ampl. São Paulo: Novatec, 2010. 608 p. ISBN 9788575222447.
- MURPHY, Mark L. **Beginning android.** New York, USA: Apress, 2009. xxii, 361 p. ISBN 9781430224198.

