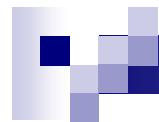




NTI / PAA PROGRAMOVÁNÍ MOBILNÍCH APLIKACÍ

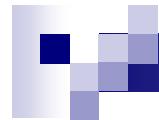
10. Broadcast, Broadcast Receiver

Ing. Igor Kopetschke – TUL, NTI
<http://www.nti.tul.cz>



Android – Broadcast

- Doposud jsme Intenty používali ke spouštění nové Activity, Service apod.
- Intenty však mohou sloužit také jako posílání anonymních zpráv mezi komponentami
 - Většinou reakce na určité změny v systému
 - **Naprosto oddělený** mechanismus od toho používaného např. při `startActivity()`
- Po odeslání broadcastu se pošle daný Intent všem BroadcastReceiverům zaregistrovaným na danou akci v Intentu



Android – BroadcastReceiver

- Zachytávání Broadcastu při komunikaci jednotlivých komponent v systému
- Je potřeba implementovat pouze metodu `onReceive()`
- Životní cyklus vázaný na danou komponentu. Jinak skončí po přijetí zprávy (`onReceive()`)
- Není vázaný na UI – nemůže vytvořit dialog, nastartovat background thread apod.

BroadcastReceiver - registrace

- Staticky v AndroidManifest.xml

```
1. <receiver  
2.     android:name=".MyReceiver">  
3.     <intent-filter>  
4.         <action  
5.             android:name="cz.cvut.example.TEST_BROADCAST" />  
6.     </intent-filter>  
7. </receiver>
```

- Nebo dynamicky v kódu (např. pokud je potřeba pouze když je activita viditelná)

```
1. IntentFilter filter = new IntentFilter(TEST_BROADCAST) ;  
2. BroadcastReceiver myReceiver = new TestReceiver() ;  
3. registerReceiver(myReceiver, filter) ; ← Provést např. v onResume()  
4. ...  
5. unregisterReceiver(myReceiver) ; ← Provést např. v onPause()
```

BroadcastReceiver

- Startují se automaticky obdržením broadcastu
- onReceive musí skončit do deseti sekund
- Pro delší operace je zde vhodné spustit službu
- BroadcastReceiver nedědí z Contextu, ten je předán metodě onReceive

```
1. public class MyReceiver extends BroadcastReceiver {  
2.  
3.     @Override  
4.     public void onReceive(Context context, Intent intent) {  
5.         int id = intent.getIntExtra("id", 0);  
6.         ...  
7.     }  
8. }
```

Broadcast - odeslání

- Broadcast můžeme odeslat dvěma způsoby
- Normal Broadcast (Non-Ordered)
 - Pošle se pomocí `Context.sendBroadcast`
 - Je kompletně asynchronní – všechny `BroadcastReceiver`y obdrží tuto zprávu v ne definovaném pořadí
 - Většinou jsou zpracovány postupně, kvůli zabránění přílišnému vytížení systému
 - Není možné ho zrušit

Broadcast - odeslání

- Ordered Broadcast
 - Posílá se pomocí `Context.sendOrderedBroadcast`
 - V jeden okamžik zpracovává zprávu maximálně jeden Broadcast Receiver
 - Pořadí se určí atributem `android:priority` v Intent filteru daného receiveru
 - Po zpracování zprávy může každý Broadcast Receiver určit, zda-li
 - Předá výsledek dalšímu Broadcast Receiveru v pořadí
 - Kompletně zruší daný broadcast a dál už se předávat nebude

Broadcast - odeslání

```
1. ...
2. public static final String TEST_BROADCAST=
3.     "cz.cvut.example.TEST_BROADCAST";
4. ...
5.
6. Intent i = new Intent(TEST_BROADCAST);
7. i.putExtra("id", 3);
8. sendBroadcast(i);
9. ...
```

- Obvykle se používá jméno balíčku, aby se zajistila jedinečnost
- Tímto řetězcem se musí zaregistrovat všechny BroadcastReceivery, které mají reagovat na tuto zprávu

Broadcast - nejpoužívanější

- Intent.*ACTION_BOOT_COMPLETED*
 - Odešle se po nastartování systému
 - Vyžaduje oprávnění RECEIVE_BOOT_COMPLETED
- Intent.*ACTION_MEDIA_BUTTON*
 - Odešle se po stisknutí tlačítka pro ovládání přehrávače (např. na sluchátkách)
 - V Intentu je v extra poli EXTRA_KEY_EVENT informace o daném tlačítku
- Intent.*ACTION_CAMERA_BUTTON*
 - Odešle se po stisknutí tlačítka fotoaparátu

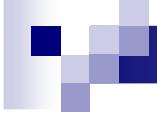
Broadcast - oprávnění

- Pro vyžádání oprávnění při odesílání broadcastu předáme nenullový parametr jedné z metod
 - `sendBroadcast(intent, receiverPermission);`
 - `sendOrderedBroadcast(intent, receiverPermission);`
- Daný broadcast přijmou pouze receivery, které mají toto oprávnění definované v *AndroidManifest.xml* pomocí `<uses-permission>`

Broadcast - oprávnění

- Pro vyžádání oprávnění při přijímání definujeme toto oprávnění při zaregistrování Broadcast Receiveru
- Dynamicky – registerReceiver(receiver, filter, broadcastPermission, scheduler);
- Staticky
 - 1. <receiver
2. android:name=".MyReceiver"
3. android:permission="android.permission.INTERNET">
4. ...
5. </receiver>
- Broadcast Receiver přijme daný broadcast pouze z aplikací, které mají toto oprávnění definované v AndroidManifest.xml pomocí <uses-permission>

Scheduler – vlákno, které má Intent přjmout



Použité a doporučené zdroje

- <http://developer.android.com/>
- <http://www.zdrojak.cz/serialy/vyvijime-pro-android/>
- <http://www.itnetwork.cz/java/android>
- <https://users.fit.cvut.cz/cermaond/dokuwiki>
- Google...



.. A to je pro dnešek vše

DĚKUJI ZA POZORNOST