



NTI/PAA - PROGRAMOVÁNÍ MOBILNÍCH APLIKACÍ

2. AndroidManifest, XML resources

Ing. Igor Kopetschke – TUL, NTI

<http://www.nti.tul.cz>

Android – Manifest

- U Javy se spouštění a konfigurace aplikace řídí manifestem. U Androidu je to podobné.
- Každá aplikace má **AndroidManifest.xml**
 - Informace o package a verzi aplikace
 - Název, ikona, theme aplikace
 - Komponenty aplikace (Activity, Service aj.)
 - Permissions (práva)
 - Minimální a maximální Android API Level
 - Použité knihovny
 - Další konfigurace
- <https://developer.android.com/guide/topics/manifest/manifest-intro>

Android – Manifest a oprávnění

- Manifest mj. řeší i oprávnění aplikace a jednotlivých komponent
- U starších verzích se při instalaci / spuštění zkontroloval úplný definovaný strom požadovaných oprávnění.
- Pokud některé z oprávnění nebylo povoleno, aplikace se nenainstaluje / nespustí
- Verze 6 má systém oprávnění přepracován.
- Aplikace může dodatečně požádat o oprávnění
- Aplikaci může být oprávnění dodatečně odmítnuto
- Řeší se tak situace, že aplikace získávala zbytečně příliš rozsáhlá práva
- <https://developer.android.com/reference/android/Manifest.permission.html>

Android manifest podrobněji

- Soubor **AndroidManifest.xml** je vždy umístěn v kořenovém adresáři aplikace.
- Manifest je povinný – bez něj aplikaci nikdy nespustíte
- Jedná se o klasický strojově čitelný XML soubor
- **<manifest>** - kořenový element s atributy
 - **xmlns:android**=<http://schemas.android.com/apk/res/android>
 - **package** – kořenový balíček aplikace (cz.tul.paa)
 - **android:versionCode** – interní číslo verze aplikace
 - **android:versionName** – veřejný identifikátor verze
 - **installLocation** – kam instalovat (většinou lze přesunout)
 - **auto** – interně, SD karta když je interní plná, rozhodnutí na systému
 - **internalOnly** – pouze interní úložiště, nelze na SD kartu
 - **preferExternal** – preferuje SD kartu pokud není plná, není garantováno

Android manifest – obsah <manifest>

- **<application>** - deklarace aplikace, viz dále
- **<uses-sdk>** - verze SDK (API Level) pro aplikaci:
 - **android:minSdkVersion** – minimální API Level
 - **android:targetSdkVersion** – API Level doporučená, když není, použije se minimální verze
 - **android:maxSdkVersion** – maximální API Level, od SDK verze 2.0.1 (API 6) ignorovaná / nedoporučovaná
- **<uses-permission>** - několik výskytů, garantuje oprávnění pro aplikaci.
 - **android:name** – identifikace oprávnění včetně package
 - **android:maxSdkVersion** – maximální API Level, pro které je oprávnění garantováno
 - Seznam oprávnění naleznete zde :
<http://developer.android.com/reference/android/Manifest.permission.html>
- **<permission>** - možnost vlastních oprávnění

Android manifest – obsah <manifest>

- **<uses-configuration>** - SW a HW požadavky:
 - **android:reqNavigation** – dpad, trackball, wheel, nonav ...
 - **android:reqTouchScreen** – notouch, stylus, finger ...
 - **android:reqKeyboardType** – nokeys, querty, twelvekey ...
 - **android:reqHardKeyboard** – true / false
 - **android:reqFiveWayNav** – true / false
- **<uses-feature>** - detailnější požadavky na HW, SW:
 - **android:name** – identifikace HW, SW
 - **android:required** – true / false
 - **android:glEsVersion** – požadovaná OpenGL ES verze
- **<compatible-screens>, <supports-screens>**
 - Tak toto je samo o sobě vypovídající ..
- Kompletní výčet – developer.android.com

Android manifest – obsah <application>

- Element s deklarací aplikace. Obsahuje jeden či více následujících subelementů:
 - **<activity>** – deklarace Aktivit s UI (viz dále.)
 - **<service>** – služby bez UI (související přednáška)
 - **<receiver>** – příjem zpráv (související přednáška)
 - **<provider>** – poskytovatelé obsahu a dat (související přednáška)
 - **<uses-library>** – výčet potřebných knihoven, má atributy
 - **android:name** – identifikace knihovny
 - **android:required** – true / false
 - **<meta-data>** – dodatečná data pro aplikaci
 - **android:name** – jednoznačné ID (klíč)
 - **android:value** – hodnota
 - NEBO
 - **android:resource** – reference na externí hodnotu

Android manifest – obsah <application>

■ Některé z atributů elementu:

- **android:allowBackup** – možnost backup / restore v rámci OS
- **android:debuggable** – lze debuggovat (true / false)
- **android:label** – název aplikace (string / resource)
- **android:description** – popis aplikace (string resource)
- **android:logo** – logo aplikace, default pro aktivitu (drawable resource)
- **android:icon** – ikona aplikace, default pro aktivitu (drawable resource)
- **android:theme** – reference na UI Theme (style resource)
- **android:supportsRtl** – podpora right-to-left (true / false)
- **android:permission** – ID oprávnění (viz předešlý slide)

■ Jak bylo řečeno dříve, element obsahuje mj. i deklarace jednotlivých aktivit aplikace

Android manifest – obsah <activity>

- Element s deklarací konkrétní aktivity.
- Aktivit může být v rámci aplikace několik a v manifestu má každá svou deklarční část
- Element má celou řadu volitelných atributů
- My si momentálně ukážeme „minimalistickou“ verzi
 - **android:name** – jméno implementující třídy
 - Včetně balíčku (**com.domain.app.MainClass**)
 - Pouze třída + relativní cesta (**.MainClass**)
 - **android:label** – popis aktivity (string / resource)
 - **android:icon** – ikona aktivity (drawable resource)
 - **<intent-filter>** – na jaký intent (záměr, akce) aktivita reaguje (viz dále)
 - **<meta-data>** – dodatečná data pro aktivitu (viz <application>)

Android manifest – <activity> <intent-filter>

- Element se může v aktivitě opakovat několikrát
- Obsahuje akci, na kterou umí aktivita zareagovat
- Volitelně obsahuje kategorii záměru a předávaná data
- Ukázka, bez které se neobejdete (hlavní aktivita)

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

Android manifest – příklad

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="cz.tul.nti.helloworld"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk android:targetSdkVersion="19" android:minSdkVersion="16" />
    <uses-permission android:name="android.permission.CALL_PHONE" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme" >
        <activity android:name=".MainActivity" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Android resources - co to vlastně je

- Aplikace v Androidu mají striktně oddělenou aplikační logiku od designu a použitých zdrojů
- Toto umožňuje vytváření aplikací pro různé velikosti obrazovek, rozlišení, jazykových verzí aj.
- Při vývoji aplikace tedy oddělujeme zdrojový kód aplikace (`*.java`) od výše uvedených zdrojů
- Tyto jsou umístěny v samostatné adresářové struktuře
- Tato struktura má kořen v `/res`

Android resources - struktura /res

- Kořenový adresář pro resources má následující strukturu
 - **drawable/** - bitmapové obrázky (png, gif, jpg)
 - **layout/** - kontejnery pro grafické komponenty (xml)
 - **menu/** - XML soubory s definicemi jednotlivých menus
 - **raw/** - jakékoli soubory, které nespádají do uvedených kategorií
 - **values/** - XML soubory s definicemi textů, stylů, polí stringů, barev, rozměrů, parametrů aj.
 - **xml/** - jakékoli další XML soubory pro aplikaci
- Kompletní seznam + význam naleznete zde:
<http://developer.android.com/guide/topics/resources/providing-resources.html>

Android resources - modifikátory

- Aplikace jsou zpravidla určeny pro více lokalizací, rozlišení, velikostí obrazovek atd.
- K tomu slouží modifikátory jako rozšíření názvů adresářů
- Priorita je dána od nejspecifičtější složky k nejobecnější
- Je taky dána pořadím uvedených modifikátorů – viz tabulka na následující stránce
- Tabulka uvádí nejčastěji užívané modifikátory.
Kompletní přehled viz zde:

<http://developer.android.com/guide/topics/resources/providing-resources.html>

Android resources - priority modifikátorů

Jazyk a region	cs, en, fr, en-rUS, en-rGB	První část udává kód jazyku, druhá (nepovinná) s prefixem r označuje region.
Dostupná šířka	w<N>dp (např.w720dp, w480dp)	Dostupná šířka se mění při změně orientace zařízení, telefon či tablet na šířku (landscape mode), dostupná šířka se oproti postavení na výšku (portrait mode) zvětší. Tento modifikátor budeme používat při tvorbě tabletových layoutů. dp je rozměrová jednotka v Androidu.
Orientace zařízení	port, land	port znamená <i>portrait mode</i> , tedy postavení na výšku, land znamená <i>landscape mode</i> , tedy na šířku.
Hustota pixelů	ldpi, mdpi, hdpi, xhdpi	Tyto modifikátory použijeme hlavně pro bitmapy, neboť aby byla reálná výsledná velikost zhruba stejná, budeme muset zařízením s jemnější hustotou pixelů dát obrázek větších rozměrů.
API level	v3, v15	Chceme-li některé resource přidělit jen zařízením s úrovní API rovnou nebo vyšší nějakému číslu, můžeme použít tento modifikátor. Pozor na to, že API 3 resp. 4 (tzn. Android 1.5 a 1.6) přijmou za svou jen verzi API té jejich se přesně rovnají.

Zdroj: <https://www.zdrojak.cz/clanky/vyvijime-pro-android-suroviny-intenty-a-jednotky/>

Android resources - jednotky

- Mobilní zařízení mívají různě velké obrazovky s různým rozlišením. Toto je potřeba zohlednit u aplikací.
- **Velikost displeje** je uváděna v palcích (inch)
- Dalším parametrem je **rozlišení** – to uvádí počet pixelů
- Z nich vychází **pixel density** (hustota pixelů, DPI, PPI). Ta uvádí **počet pixelů na 1 palec**.
- Má-li tedy displej 320DPI, znamená to, že má 320 pixelů na palec
- Jak z toho chaosu tedy ven ?

Android resources - jednotky

- Android proto nepoužívá pro uvádění rozměrů klasický pixel
- Virtuální jednotka **Density-independent pixel (dp)**
- Definována jako **1dp = 160px / dpi**
- Používá se při definici rozměrů jednotlivých View
- A jak je to s bitmapami ?
- Toto řeší modifikátory (viz předešlý text), které definují základní velikosti displejů, jejich dpi a poměry mezi nimi
- Viz další strana ...

Android resources - modifikátory pro displej

- Velikost obrazovky (screen size)
 - **small** – 320 x 426 dp
 - **normal** – 320 x 470 dp
 - **large** – 480 x 640 dp
 - **xlarge** – 720 x 960 dp
- Hustota pixelů dpi (poměr 3:4:6:8:12:16)
 - **ldpi** – 120 dpi (uložený obrázek má 15x15 px)
 - **mdpi** – 160 dpi (uložený obrázek má 20x20 px)
 - **hdpi** – 240 dpi (uložený obrázek má 30x30 px)
 - **xhdpi** – 320 dpi (uložený obrázek má 40x40 px)
 - **xxhdpi** – 480 dpi (uložený obrázek má 60x60 px)
 - **xxxhdpi** – 640 dpi (uložený obrázek má 80x80 px)

Android resources – definice, použití

- Nyní si blíže představíme jednotlivé zdroje
- A nezapomeneme ani na způsob, jak k nim přistupovat
- Zdroje jsou primárně umístěny v adresářích dle typu (viz pár stránek zpět)
- Obecné umístění:
 - `res/res_type/filename.xml`
- Přístup k nim:
 - Java – `R.res_type.key_id`
 - XML – `@res_type/key_id`

res/values/strings.xml

- Definice řetězců pro aplikaci a manifest

```
<resources>  
    <string name="hello">Ahoj</string>  
    <string name="home">Domů</string>  
</resources>
```

- Přístup k nim:

- ☐ Java – `text = getString(R.string.hello) ;`
- ☐ XML – `android:text = "@string/hello"`

res/values/strings.xml

- Definice polí řetězců pro aplikaci

```
<resources>
    <string-array name="days">
        <item>Pondeli</item>
        <item>@string/utery</item>
    </string-array>
</resources>
```

- Přístup k nim:

- ☐ Java – `Resources res = getResources();`
`res.getStringArray(R.array.days);`
- ☐ XML – není

res/values/strings.xml - lokalizace

■ /res/values-cs/strings.xml

```
<resources>
    <string name="hello">Ahoj</string>
    <string name="home">Domů</string>
</resources>
```

■ /res/values-en/strings.xml

```
<resources>
    <string name="hello">Hello</string>
    <string name="home">Home</string>
</resources>
```

res/values/bools.xml

- Definice logických hodnot

```
<resources>
    <bool name="allow_write">true</bool>
    ...
</resources>
```

- Přístup k nim:

- Java – `getBoolean(R.bool.allow_write) ;`
- XML – `"@bools/allow_write"`

res/values/colors.xml

- Definice pojmenovaných barev

```
<resources>  
    <color name="moje">#80ff0000</color>  
    ...  
</resources>
```

- Přístup k nim:

- ☐ Java – `getColor(R.color/moje)` ;
- ☐ XML – `"@color/moje"`

res/values/dimens.xml

- Definice pojmenovaných rozměrů

```
<resources>
    <dimen name="siroke">250dp</dimen>
    ...
</resources>
```

- Přístup k nim:

- ☐ Java – `getDimension(R.dimen/siroke)` ;
- ☐ XML – `"@dimen/siroke"`

res/values/integers.xml

- Definice pojmenovaných celočíselných hodnot

```
<resources>
    <integer name="max_speed">250dp</integer>
    ...
</resources>
```

- Přístup k nim:

- ☐ Java – `getInteger(R.integer/max_speed)` ;
- ☐ XML – `"@integer/max_speed"`

- Integer Array - analogicky jako u stringů

res/values/arrays.xml

- Definice polí z jiných zdrojů

```
<resources>
    <array name="icons">
        <item>@drawable/home</item>
        <item>@drawable/logout</item>
    </array>
</resources>
```

- Přístup k nim:

- ☐ Java – `Resources res = getResources();`
`res.obtainTypedArray(R.array.icons);`
- ☐ XML – není

res/menu/some_menu.xml

- Definice konkrétního menu

```
<menu>
    <item android:id=@+id/home
          android:title=@string/home/>
    <item android:id=@+id/help
          android:title=Pomoc/>
</menu>
```

- Přístup k nim:

- ☐ Java – `R.menu.some_menu`
- ☐ XML – `@menu/some_menu`

- **<item>** má spoustu dalších atributů

- ☐ icon (drawable)
- ☐ onClick (metoda)
- ☐ visible, enabled
- ☐ aj.

Android resources – layout, drawable ...

- Definice UI pro aktivitu
- Podrobný popis vyžaduje nejdříve znalost tříd Activity a View / ViewGroups
- Logicky se nejdříve zaměříme na ty, které nám umožní vyvíjet jednoduché aplikace na cvičení
- Cvičení tedy budou zároveň doplňovat přednášky, protože praktické užití je za tisíce teoretických slov
- Doporučuji paralelně s výukou navštívit stránky

<http://developer.android.com>



.. A to je pro dnešek vše

DĚKUJI ZA POZORNOST