

Week 4 - oefening 2: Intent Filters

Doelstelling

In dit labo onderzoek je het gebruik van implicit intents, intentfiltering,...

Voorbereiding

Maak een nieuw Android Application Project aan met volgende instellingen.

Application Name:	Demo Intent Filters
Package Name:	be.howest.nmct.intentfiltersproject
Activity/fragment:	MainActivity & MainFragment
Target SDK:	21 (of eerder)

Design

De gevraagde (aan te maken) activiteiten vragen geen bijzonder design.

Code

Ga zelf even op onderzoek uit door onderstaande vragen te beantwoorden. De achtergrond (en te kennen leerstof) is te vinden op:

<http://developer.android.com/training/basics/intents/sending.html>

<http://developer.android.com/guide/components/intents-filters.html>

Onderzoek 1: bestudeer het AndroidManifest-file.

Maak een nieuwe java klasse 'Constants' aan en plaats hierin de constante:

```
public static final String ACTION_IMPLY="be.howest.nmct.intentfiltersproject.intent.action.IMPLY";
```

Plaats in de layout MainActivity een Button met daaraan de methode launchWithAction

```
public void launchWithAction(View v) {  
    Intent intent = new Intent(Constants.ACTION_IMPLY);  
    startActivity(intent);  
}
```

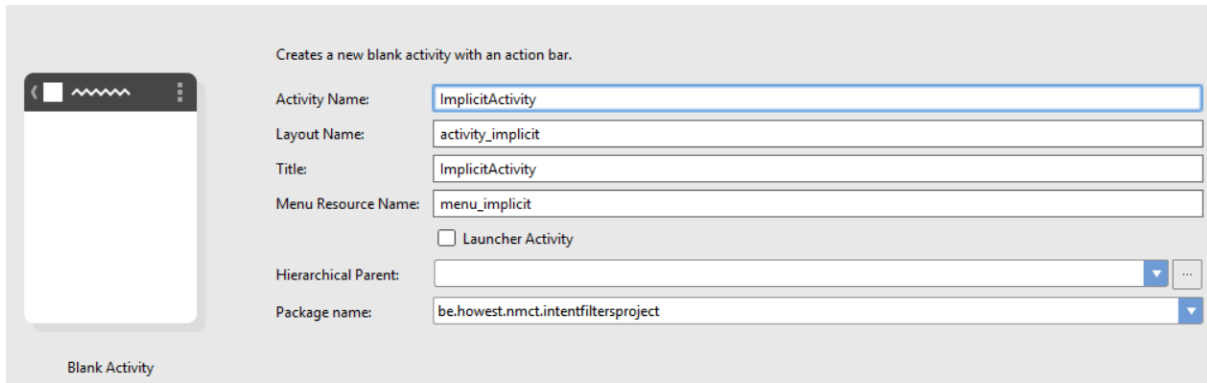
Wat merk je? Welke exception(s) krijg je te zien?

Pas de code zodat je een Toast message ziet als de Intent niet bestaat. Maak gebruik van queryIntentActivities. Baseer je op:

<http://developer.android.com/training/basics/intents/sending.html#Verify>

Onderzoek 2:

Maak een nieuwe Blank Activity 'ImplicitActivity' aan met volgende parameters:



Voeg in het AndroidManifest-file een Intentfilter toe voor deze nieuwe activity

```
<activity
    android:name="be.howest.nmct.intentfiltersproject.ImplicitActivity"
    android:label="@string/title_activity_implicit" >

    <intent-filter>
        <action android:name="be.howest.nmct.intentfiltersproject.intent.action.IMPLY" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>

</activity>
```

Opgelet: zorg ervoor dat de vermelde stringwaarde bij action-tag identiek is met de stringconstante vermeld in Constants.java!

Vragen:

- Deploy de activity opnieuw en druk op de knop. Wat gebeurt nu?
- Wat gebeurt er als we de category tag in commentaar zetten? Ga in de documentatie na waarvoor de Default category dient? Is hij altijd nodig?
- Blijft de toepassing werken indien we een tweede category (in het AndroidManifest-file) toevoegen met de naam android.intent.category.LAUNCHER?

Pas jouw code nu als volgt aan:

```
Intent intent = new Intent(Constants.ACTION_IMPLY);
intent.addCategory(Intent.CATEGORY_CAR_DOCK);
```

Vragen:

- werkt nu de methode nog? Waarom niet?
- Pas het AndroidManifest-file zodat de methode opnieuw werkt.

Beschrijf nu zelf hoe Android omgaat met categoriën in de intent en de intentfilter om te beslissen of een activity al dan niet voldoet.

Zie ook op:

<http://developer.android.com/guide/components/intents-filters.html#Resolution>

Onderzoek 3:

Van waar komt de waarde die in het android:name attribuut van de action staat?

Blijft de toepassing werken indien we in het AndroidManifest-file een tweede action met de name android.intent.action.VIEW toevoegen?

Wat gebeurt er als we de code nu als volgt aanpassen? Hoe komt dit?

```
Intent intent = new Intent("android.intent.action.VIEW");
startActivity(intent);
```

Onderzoek 4:

Pas de IntentFilter in het AndroidManifest.xml bestand als volgt aan:

```
<activity
    android:name="be.howest.nmct.intentfiltersproject.ImplicitActivity"
    android:label="@string/title_activity_implicit" >
    <intent-filter>
        <action android:name="be.howest.nmct.intentfiltersproject.intent.action.IMPLY" />

        <data android:scheme="xtp" />

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

Voeg in de MainActivity een extra button toe en koppel aan die button volgende methode. Deze methode ziet er als volgt uit:

```
private void onderzoek4() {
    //Intent intent = new Intent(Constants.ACTION_IMPLY, Uri.parse("xtp:///somedata"));
    //startActivity(intent);

    // test zonder data
    Intent intent = new Intent(Constants.ACTION_IMPLY);
    startActivity(intent);
}
```

Wat werkt? Verklaar nader. Welke attributen heeft de data-tag nog en waarvoor dienen ze? Merk op dat er met de data in de tweede activity nog niets gedaan wordt.

Meer info terug te vinden op:

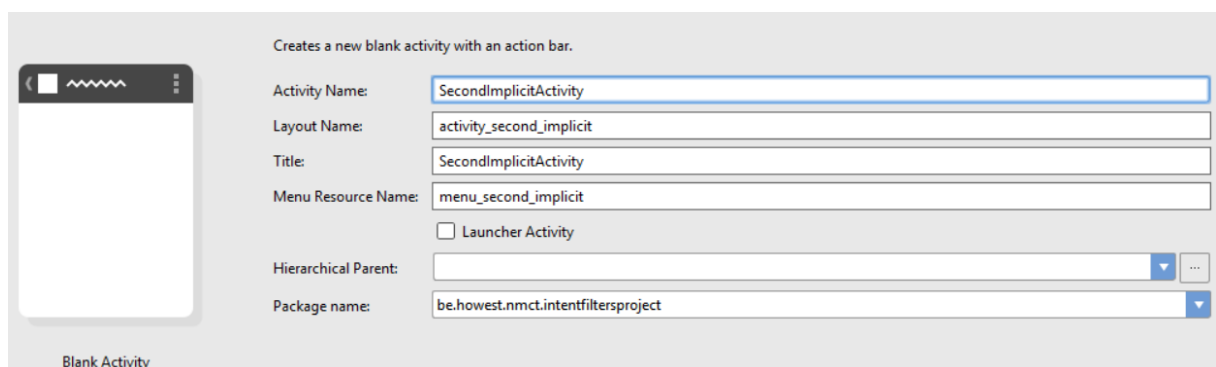
<http://developer.android.com/guide/components/intents-filters.html#Resolution>
<http://developer.android.com/guide/topics/manifest/data-element.html>

Pas bovenstaande code aan:

```
Intent intent = new Intent(Constants.ACTION_IMPLY, Uri.parse("xtp:///somedata"));
startActivity(intent);
```

Onderzoek 5:

Maak een nieuwe Blank Activity 'SecondImplicitActivity' aan met volgende parameters



Zorg dat deze (in het AndroidManifest.xml) dezelfde IntentFilter als ImplicitActivity heeft. Wat gebeurt er nu als je op de tweede button duwt? Hoe gaat android hiermee om?

Onderzoek 6:

Pas de IntentFilter in het AndroidManifest.xml bestand als volgt aan. Surf nu met de browser op je toestel naar <http://www.kortrijk.be> en klik op nieuws. Wat gebeurt er? Blijft dit gebeuren als de BROWSABLE categorie er niet is? Wat zegt de documentatie over BROWSABLE? Zijn er nog zulke speciale categoriën?

```
<activity
    android:name="be.howest.nmct.intentfiltersproject.SecondImplicitActivity"
    android:label="@string/title_activity_second_implicit" >

    <intent-filter>
        <action android:name="android.intent.action.VIEW" />

        <category android:name="android.intent.category.DEFAULT" />
        <category android:name="android.intent.category.BROWSABLE" />

        <data
            android:host="www.kortrijk.be"
            android:path="/nieuws"
            android:scheme="http" />
    </intent-filter>
</activity>
```