Why does intent.resolveActivity(getPackageManager()) return null even though there is Activity to handle it?

Asked 11 months ago   Active 18 days ago   Viewed 2k times

I'm following the [Android Codelabs](#), specifically I'm working on [this](#) Codelab with implicit intents.

The Codelab has the following method:

```
public void openWebsite(View view) {
    String url = mWebsiteEditText.getText().toString();

    Uri webpage = Uri.parse(url);
    Intent intent = new Intent(Intent.ACTION_VIEW, webpage);

    if (intent.resolveActivity(getPackageManager()) != null) {
        startActivity(intent);
    } else {
        Log.d("ImplicitIntents", "Can't handle this intent!");
    }
}
```

The problem is that the `intent.resolveActivity(getPackageManager())` returns null, but if I omit this and just call the `startActivity(intent)`, it works fine and opens the Uri in Google Chrome.

I'm wondering why `intent.resolveActivity(getPackageManager())` returns null, even thought the Uri can be opened in Google Chrome?

The **Manifest** file:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.implicitintents">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        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>
```

In this case the URL that we want to open comes from `EditText` field, so we can't use an `intent-filter` with `android:host` as described [here](#).

Share Improve this question Follow

asked Aug 11 '20 at 12:27

Antonis Papadakis
1,130 ● 1 ● 9 ● 22

If you are testing on Android 11 or higher, you are being affected by package visibility restrictions. – CommonsWare Aug 11 '20 at 12:34

@CommonsWare You are right. I tested it in the previous version and it works. I'm searching how can this be solved in Android 11. – Antonis Papadakis Aug 11 '20 at 13:16

3 Answers

Active   Oldest   Votes

2

In case you haven't solve the problem yet. I don't know if you are using API level 30, but if you are you should check this new Package visibility restrictions. Package Visibility has changed and apps are no longer able to access the Package Manager directly. You need to add a element in your AndroidManifest file. In your case you would need something like this.

```
<queries>
    <intent>
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.BROWSABLE" />
        <data android:scheme="https" />
    </intent>
</queries>
```

Share Improve this answer Follow

edited Nov 16 '20 at 20:05

Dharman
22.3k ● 19 ● 58 ● 108

answered Nov 16 '20 at 20:00

abrahamvee
23 ● 3

1

I had the same problem. I solved it by adding a request to use the browser in the manifest. Now my manifest file looks like this. Try reading this article.

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.android.quakereport">
<queries>
    <intent>
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.BROWSABLE" />
        <data android:scheme="https" />
    </intent>
</queries>
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".EarthquakeActivity">
        <intent-filter>
```

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

Share Improve this answer Follow

add permission

0

```
<uses-permission android:name="android.permission.QUERY_ALL_PACKAGES" />
```

then add Queries

```
<queries>
    <intent>
        <action android:name="android.intent.action.VIEW" />
        <data android:scheme="https" />
    </intent>
</queries>
```

I hope it will solve your problem. you can also check package visibility use cases

Share Improve this answer Follow