AndroidGPSBlog

How difficult is it to turn an Android Phone into a fully fledged GPS unit? Starting from scratch, this blog illustrates Android programming for a medium-complex real-world example.

Wednesday, 24 September 2008

Starting an Android Service at Boot time

A service that has to be started manually is an oxymoron, so starting a service at boot time is for many applications a must.

My tracklogging service is such an example, which should run whenever the phone is turned on, so that one can refer to the route travelled later. A further example is the much touted CarbonFootprint, which again relies on accurate measurements all the time, and not just when the user has turned it on.

The last post detailed how the TrackloggingService worked, but started the service only when the main activity was launched. Now comes the time to hook it into the Android boot sequence. Here is how:

After boot completes the Android system broadcasts an intent with the action android.intent.action.BOOT_COMPLETED. And now all we need is an IntentReceiver, now called a BroadcastReceiver, to listen and act on it. This is how this class looks:

```
public class LocationLoggerServiceManager extends BroadcastReceiver {

 public static final String TAG = "LocationLoggerServiceManager";
 @Override
 public void onReceive(Context context, Intent intent) {
  // just make sure we are getting the right intent (better safe than sorry)
  if( "android.intent.action.BOOT_COMPLETED".equals(intent.getAction())) {
   ComponentName comp = new ComponentName(context.getPackageName(), LocationLoggerS
ervice.class.getName());
   ComponentName service = context.startService(new Intent().setComponent(comp));
   if (null == service){
    // something really wrong here
    Log.e(TAG, "Could not start service " + comp.toString());
   }
  } else {
   Log.e(TAG, "Received unexpected intent " + intent.toString());
  }
 }
}
```

The key is of course the onReceive() method. I have decided to check that it is actually the Intent I am expecting, but otherwise it is straightforward starting the service and return.

The receiver needs to be declared in the manifest, e.g. with the following entry:

```
<receiver android:name=".LocationLoggerServiceManager"
 android:enabled="true"
 android:exported="false"
 android:label="LocationLoggerServiceManager">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED" />
    </intent-filter>
</receiver>
```

Furthermore this class listen to this specific event needs to be declared in the security settings:

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

That's it. Now the service will be started as soon a Android has finished booting.
Now we will need something to make this user-configurable...

Ludwig at 20:19

8 comments:

jeremy.org 8 December 2008 16:37

Thanks! This was useful.

Reply

cryptyk 17 January 2009 18:14

This was very helpful. It got me unstuck.

Thank you.

Reply

Abhi 7 January 2010 12:11

Hi,

I am getting the following error when i added your code

Error : "The application has stopped unexpectedly please try again"

Reply

Ludwig 7 January 2010 12:17

Without more information I do not think anyone will be able to help you. It looks like the code worked for others. It might be best to have a look in the debugger what is happening in your code. I assume it is some form of ActivityNotFound exception...

Reply

AF-SH-ISH 30 April 2010 10:23

This comment has been removed by the author.

Reply

leo 16 August 2011 07:47

thanks you! And I have another question. How can I start the "LocationLoggerServiceManage" or need to register it?

Reply

leo 16 August 2011 07:53

Thanks. I'm new to andriod and JAVA. I want to know how to start the "LocationLoggerServiceManager". Thanks in advance.

Reply

herri 15 March 2012 22:08

This seems to be working in a emulator, but on an actual device it does not. I tested the code on both, 2.1 and 2.2 ZTE Blade Android handsets.

The other device requests for a PIN code and the other does not have SIM card in it, so it cannot be that the signal broadcasted would be delayed, causing a timeout or something.

Hopefully someone will get the code fixed?

Reply

Enter your comment...

Comment as: Google Accour ▾

Publish Preview

‹ Home ›

View web version