HapticGuide tutorial

The HapticGuide is a module which helps you guiding someone to a GPS location. In this tutorial we will create a small program which saves a GPS location and then guides the user back to this location. We call this application HelloGuide. After creating the new Android project (we used Android 2.2) you have something like this in helloGuide.java:

```java
package se.lth.certec.helloguide;

import android.app.Activity;
import android.os.Bundle;

public class HelloGuide extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Edit the layout xml to include two buttons:

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:id="@+id/textView1"
    android:text="HapticGuideTest"
           />
<Button android:layout_width="fill_parent" android:text="Set location"
android:layout_height="wrap_content" android:id="@+id/set"></Button>
<Button android:layout_width="fill_parent" android:text="Guide me!"
android:layout_height="wrap_content" android:id="@+id/guide"></Button>
</LinearLayout>
```

Add buttons and click listeners to HelloGuide.java:

```java
package se.lth.certec.helloguide;

import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;


public class HelloGuide extends Activity {
    private Button button1;
    private Button button2;
    private final String TAG = "MyActivity";

    @Override
    public void onCreate(Bundle savedInstanceState) {
           super.onCreate(savedInstanceState);
           setContentView(R.layout.main);
           button1 = (Button) findViewById(R.id.set);
           button1.setOnClickListener(new OnClickListener() {
                      public void onClick(View v) {
                                 Log.i(TAG, "set button");
                      }
           });
           button2 = (Button) findViewById(R.id.guide);
           button2.setOnClickListener(new OnClickListener() {
                      public void onClick(View v) {
                                 Log.i(TAG, "guide button");
                      }
           });
    }
}
```

Now it is time to grab a location. Right click on the HelloGuide project, select properties, Android and add the HaptiMap toolkit in the Library section. Add

```
import org.haptimap.hcimodules.util.MyLocationModule;
```

and also

```
import android.widget.Toast;
```

to HelloGuide.java (toast will be used to provide a bit of feedback for the user). To get a location add:

```
           private MyLocationModule myLocation;
           private Location currentPos;
```

To the Hello Guide class, and start myLocation and set currentPos to null in onCreate:

```
           myLocation=new MyLocationModule(this);
           myLocation.onStart();
           currentPos=null;
```

Add

```
currentPos = myLocation.getCurrentLocation();
if (currentPos==null){
           Toast.makeText(HelloGuide.this, "no GPS signal - no position
           set", Toast.LENGTH_SHORT).show();
} else {
           Toast.makeText(HelloGuide.this, "location set= " +
           currentPos.getLatitude() + ", " + currentPos.getLongitude(),
           Toast.LENGTH_SHORT).show();
}
```

To the button1 onClick function. Since we added& started the MyLocationModule we also have to take care to stop/destroy it. Add:

```
@Override
protected void onDestroy() {
            myLocation.onDestroy();
            super.onDestroy();
}
```

There is one more step that has to be done before this program will run properly (if you try to run it like this it will report it has to stop unexpectedly). Open the AndroidManifest file and select the tab **permissions**. Press Add, and select Uses Permossion. In the drop down to the right, select **android.permission.ACCESS_FINE_LOCATION**. Save. Now the application should be ready to run. If you are indoors you will not have access to the GPS, but the "My Fake Location" app (which allows faking a gps signal) can be helpful if you don't want to go outside.

Now that we can save a position, we also want to be guided back there. Add

```
private HapticGuide theGuide;
```

to the class. This requires the following imports:

```
import org.haptimap.hcimodules.guiding.HapticGuide;
import org.haptimap.hcimodules.util.WayPoint;
```

Create it in onCreate:

```
theGuide=new HapticGuide(this);
```

Now the only thing we have to do is to make sure we have a valid position and then we can start theGuide. Add the following in the button2 onClick function:

```
if (currentPos!=null){
            WayPoint goal= new WayPoint("goal", currentPos);
            theGuide.setNextDestination(goal);
            theGuide.onStart();
} else {
            Toast.makeText(HelloGuide.this, "no GPS signal - cannot guide",
            Toast.LENGTH_SHORT).show();

                                                       }
```

Also remember to take care of the destruction. Add

```
theGuide.onDestroy();
```

in onDestroy. Before running it there is a permission to set (otherwise the guiding won't work). Go to the AndroidManifest, Permissions again and add Uses Permission **android.permission.VIBRATE**.

In principle we could stop here, but we will add one final touch that allows you to add more information by registering a listener to the HapticGuide:

```
theGuide.registerHapticGuideEventListener(new HapticGuideEventListener(){

            public void onRateIntervalChanged(int millis) {

            }

            public void onPrepared(boolean onPrepared) {

            }

            public void onDestinationReached(long[] pattern) {
                        Toast.makeText(HelloGuide.this, "You have arrived!",
                        Toast.LENGTH_SHORT).show();

});
```

That is it – time to go out and test!