

```

1 public class ShareMyPosition extends MapActivity implements LocationListener {
2
3     private LocationManager locationManager;
4
5     public void onCreate(Bundle savedInstanceState)
6     { ...locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE); ...}
7
8     private void performLocation(boolean forceNetwork)
9     {
10         ...List<String> providers = locationManager.getProviders(true); ...
11         boolean containsGPS = providers.contains(LocationManager.GPS_PROVIDER);
12         boolean containsNetwork = providers.contains(LocationManager.NETWORK_PROVIDER);
13         if ((containsGPS && !forceNetwork) || (containsGPS && !containsNetwork)) {
14             locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 5, this);
15         } else {...finish();}
16
17     protected void onResume()
18     {
19         ... performLocation(false); }
20
21     protected void onPause()
22     {
23         ... locationManager.removeUpdates(this); }
24
25     protected Dialog onCreateDialog(int id){
26         return new AlertDialog.Builder(this).setTitle(R.string.app_name)
27         .setView(sharedMapView)
28         .setNeutralButton(R.string.options, new OnClickListener() {
29             public void onClick(..., ...)
30             {...startActivity(option);} })
31         .setPositiveButton(R.string.share_it, new OnClickListener() {
32             public void onClick(..., ...)
33             {...startActivity(share);} })
34         .setNegativeButton(R.string.retry, new OnClickListener() {
35             public void onClick(DialogInterface arg0, int arg1)
36             {...performLocation(false);}
37         }).create()
38
39     public void onLocationChanged(final Location location)
40     {
41         ...locationManager.removeUpdates(this);
42         this.location = location;...}}

```