Integrating a Location Provider - Location SDK - Android

This documentation is no longer actively supported and may be out of date. Going forward, please visit and bookmark our new site (https://docs.phunware.com/) for up-to-date documentation.

(v3.0.0-3.1.0 of Location SDK-Android)

Integrating a Location Provider

This guide provides instructions for integrating Location SDK for routing. It is only applicable for users of the Android Location SDK v 3.0.0-3.1.0.

*If you are using a more recent version of the Android Location SDK, view the Location SDK-Managed Provider Integration Guide.

*If you are using an earlier version of the Location SDK for Android contact Phunware Support (support@phunware.com).

In order to show the user's current location, you must add a Location Provider, which updates location information.

Phunware's location providers:

Provider	Settings/Keys	Compile Statement	Description
BLE	• Confidence Factor • Acceptance Radius • Map Key • Customer ID	`com.phunware.location:provider-senion:3.0.0`	Senion Lab BLE (bluetooth low energy) location provider
CMX - Hyperlocation (MSE)	• Confidence Factor • Venue GUID • Minimum Stationary Time • Maximum Lingering time	`com.phunware.location:provider-cmx:3.0.0`	Cisco Hyperlocation (wifi) location provider
GPS		`com.phunware.location:provider-gps:3.0.0`	GPS location provider

Step 1 - Add a location provider as a compile dependency

```
apply plugin: 'com.android.application'
android {
    ...
}

dependencies {
    ...
    compile
'com.phunware.mapping:mapping:3.1.2'
    compile
'com.phunware.location:provider-senion:3.1.0'
    ...
}
```

Step 2 - Set the Location Provider and Enable Location Updates

This step tells the PhunwareMapManager what provider will be used to get location updates. In this case, the Senion Location Provider requires a customer id and map id. The method getSenionFloorMap() returns a HashMap of Senion floor ids to MaaS floor ids.

It's important to note that you must pass a building object to the setLocationManager() method on PhunwareMapManager.

The call to setMyLocationEnabled() allows you to control when the blue dot is rendered.

```
@Override
public void onPhunwareMapReady(PhunwareMap
phunwareMap) {
 mapManager.setPhunwareMap(map);
 mapManager.addBuilding(getResources().getInteg
er(R.integer.building_id),
        new Callback<Building>() {
            @Override
            public void onSuccess(final
Building building) {
    ...


mapManager.setLocationProvider(SenionProviderFa
ctory.create(this,

getString(R.string.sl_customer_id),

getString(R.string.sl_map_id),

getSenionFloorMap()).createLocationProvider(),
building);

                // enable my location (blue
dot)

mapManager.setMyLocationEnabled(true);


    ...
            }

            @Override
            public void onFailure(Throwable e)
{
            }
        });
}
```

Step 3 - Manage Location Updates when in the Background

Constant location updates when in the background will drain a battery. In order to prevent this, we intercept activity lifecycle methods to disable and enable location updates.

```
@Override
protected void onPause() {
    super.onPause();
    if (mapManager != null) {
        mapManager.setMyLocationEnabled(false);
    }
}

@Override
protected void onResume() {
    super.onResume();

    if (mapManager != null) {
        mapManager.setMyLocationEnabled(true);
    }
}
```