Garmin Health SDK

*Android Version 2.3.0 Migration Guide*

The transition from 2.2.X to 2.3.X versions of the SDK introduces one major new feature, native support for initializing the Health SDK as an Android service. The following guide is designed to aid partners in migrating from existing implementations, for a full guide to working with the Health SDK Service please review the *ServiceConfiguration.html* documentation bundled with the SDK.

**Basic Configuration**

For partners who don’t wish to implement the Health SDK as a service, the only change that will be required is a single additional to the application build.gradle file.

If you do not wish to leverage the Health SDK service, then no changes need to be made. If you do intend to work with the Service then add the following configuration to the Application tag of your AndroidManifest.xml file:

<service android:name="com.garmin.health.GarminHealthService"  
 android:enabled="true"  
 android:process=":<YOUR PROCESS NAME HERE>"  
 tools:replace="android:enabled,android:process"/>  
<service android:name="com.garmin.android.gncs.GNCSListenerService"  
 android:process=”:<YOUR PROCESS NAME HERE>"  
 tools:replace="android:process"/>

**SDK Initialization**

If your solution is designed to use the Health SDK Service, then please listen for the completion of the *ListenableFuture* returned by the *GarminHealth#initialize()* method before attempting SDK communication. For an example see below.

public class MainActivity extends AbstractGarminActivity  
{  
 protected void onStart()  
 {  
 super.onStart();  
   
 Futures.addCallback(HealthSDKManager.initializeHealthSDK(this),

new FutureCallback<Boolean>()  
 {  
 @Override  
 public void onSuccess(@Nullable Boolean result)  
 {  
 // Start working with the Health SDK...  
 }  
   
 @Override  
 public void onFailure(@NonNull Throwable t)  
 {  
 // Handle initialization failure...

finishAndRemoveTask();  
 }  
 }, Executors.newSingleThreadExecutor());  
 }  
}

**API Changes**

Under rare circumstances the Health SDK service can fail while your application is still running. The majority of Health SDK methods already return types, such as *ListenableFuture*, that can handle this failure. One existing SDK method *DeviceManager#getPairedDevices()* has been declared Nullable in version 2.3.0. Null will only be returned if the SDK Service is being used, but is not active. Please sanitize your use of this API method to ensure that potential null returns do not crash your application.

A number of listener interfaces in the Health SDK have been changed to include a defaulted callback *onServiceDisconnected()*.

DevicePairStateListener

LoggingSyncListener

SyncListener

DeviceConnectionStateListener

RealTimeDataListener

This callback should signal to your listener implementation that it will not receive additional communication callbacks until the Health SDK service has been reinitialized. Listeners that include this new callback include:

**Please Review the ServiceConfiguration.html documentation file for more details on using the Health SDK Service in your implementation.**

If you have any questions or concerns regarding the new SDK build, please contact Garmin Health SDK Support at [sdksupport@health.garmin.com](mailto:sdksupport@health.garmin.com)