

VCS SDK 使用说明文档

1 依赖配置

拿到VCS的SDK: **swap-xxx.jar** , 放到模块的**libs**目录下, 并在模块的**build.gradle**中增加依赖

```
dependencies {  
    .....  
    implementation files('libs/swap-xxx.jar')  
    .....  
}
```

2 swap.jar 使用说明

新建Service, 并在 AndroidManifest.xml中注册

注册Service类

```
<application>  
    .....  
    <service  
        android:name=".XxxService"  
        android:enabled="true"  
        android:exported="true" />  
    .....  
</application>
```

新建Service类

```

public class XxxService extends Service {
    private static final String TAG = "XxxService";

    //vcbNotificationManager 用于主动通知VCS事件
    private VcbNotificationManager mVcbNotificationManager;

    //mBridgeListeners 用于保存客户端注册的监听器，用于异步返回给客户端操作结果
    private final Set<IBridgeListener> mBridgeListeners = new HashSet<>();

    // 实现aidl stub
    private final IBridge.Stub mBind = new IBridge.Stub() {
        @Override
        public String sendRequest(String uuid, String msg) throws RemoteException {
            //接收VCS发来的json指令。
            // 可以同步返回结果：组装成json返回；
            // 不能同步处理结果：返回null，然后通过listener通知处理结果；
            xxxFunction(uuid);
            return null;
        }

        @Override
        public void registerBridgeListener(IBridgeListener bridgeListener) throws RemoteException {
            mBridgeListeners.add(bridgeListener);
        }
    };

    private void xxxFunction(final String uuid) throws RemoteException {
        //处理数据，调用接口，返回结果或通过listener回调给VCS
        String callbackMsg = "";

        for (IBridgeListener listener : mBridgeListeners) {
            if (null != listener) {
                // 操作成功调用 onResponse
                listener.onResponse(uuid, callbackMsg);
                // 操作失败调用 onFailure
                listener.onFailure(uuid, callbackMsg);
            }
        }
        // 这里如果需要播报TTS，或其他事件通知到VCS，调用下面方法
        String notifyUuid = UUID.randomUUID().toString();
        String notifyEvent = ""; //组装好的json字符串，格式待定义
        if (null != mVcbNotificationManager){
            mVcbNotificationManager.notifyEvent(notifyUuid, notifyEvent);
        }
    }

    @Override
    public void onCreate() {
        super.onCreate();
        mVcbNotificationManager = new VcbNotificationManager(this);
    }
}

```

```

@Override
public IBinder onBind(Intent intent) {
    return mBind;
}

@Override
public boolean onUnbind(Intent intent) {
    return super.onUnbind(intent);
}

@Override
public void onDestroy() {
    super.onDestroy();
}
}

```

3 swap.jar 接口说明

3.1 IBridge.Stub 接口说明

接口名称：客户端发送请求控制、查询...

String sendRequest(String uuid, String msg)

接口参数

uuid：标识此次请求的唯一识别码

msg：json格式指令字符串

返回值：

null 或 json 格式字符串

null表示此次请求无法同步返回给客户端，需要客户端通过监听器接收指令执行结果
json格式字符串表示此次请求的请求结果，例如：

```

{
    "code":0,
    "msg":"操作成功"
}

```

具体格式根据不同请求返回不同格式

3.2 IBridgeListener 接口说明

接口名称：服务端指令执行成功返回响应

```
void onResponse(String uuid, String msg)
```

接口参数

uuid：返回码，需要和请求码一致

msg：额外信息，可以为null、string、json格式字符串

接口名称：服务端指令执行失败返回响应

```
void onFailure(String uuid, String msg)
```

接口参数

uuid：返回码，需要和请求码一致

msg：额外信息，可以为null、string、json格式字符串

3.3 VcbNotificationManager 接口说明

接口名称：服务端主动通知事件接口

```
void notifyEvent(@NonNull String uuid, String event)
```

接口参数

uuid：事件唯一标识码

event：事件信息，json格式

4 Demo示例

Demo 工程请参考附件：xxxservicedemo.zip