

Android SDK集成文档

一. 导入SDK

在主模块目录下的libs目录（如果没有libs目录则创建libs目录）下导入kssdklibrary.aar



二. 快速集成

1、项目build.gradle中添加repositories

```
repositories {
    flatDir {
        dirs 'libs'
    }
}
```

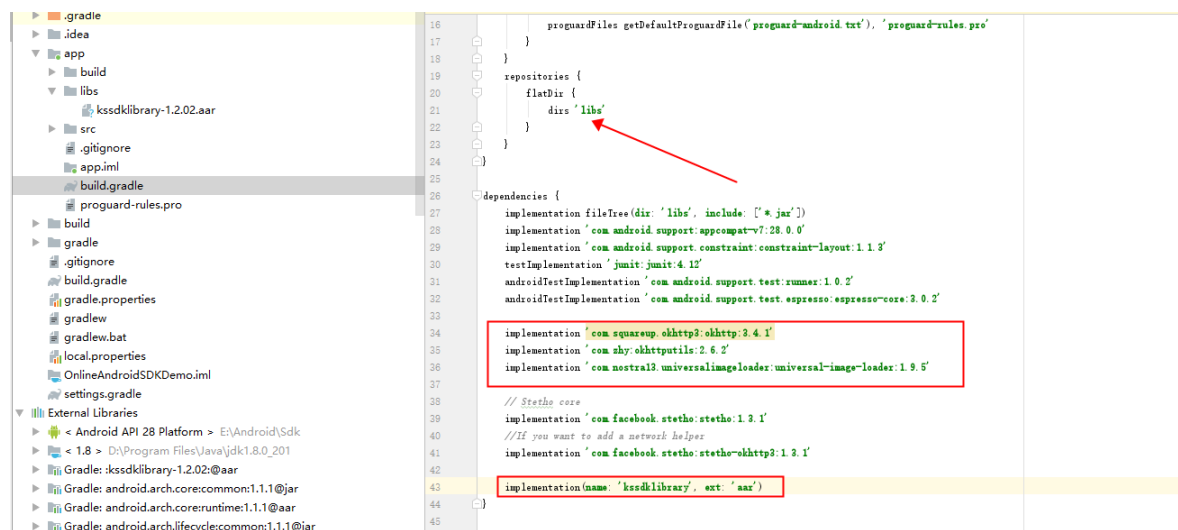
2.项目build.gradle中添加dependencies:

```
compile 'com.squareup.okhttp3:okhttp:3.4.1'

compile 'com.zhy:okhttputils:2.6.2'

compile 'com.nostra13.universalimageloader:universal-image-loader:1.9.5'

compile (name: 'kssdklibrary', ext: 'aar')
```



2.初始化kssdk，其中appKey需要修改为自己配置的字符串，companyId为公司ID

```

KSConfig.init(this, "T/z+xakyDBMvBlWPuiQm8JqEaKovIDK9", "xxxxxx", new
KSInitListener(){
    @Override
    public void onSuccess() {
        //TODO 初始化成功
    }

    @Override
    public void onError(int code, String message) {
        //TODO 初始化失败
    }
});

```

3.打开咨询窗口，只有初始化成功后，才能正常打开

```

Intent intent = new KSIntentBuilder(MainActivity.this).build();
startActivity(intent);

```

4.主动关闭对话(比如退出app时)

```

KSConfig.closeDialog(MainActivity.this);

```

三、电商功能

该功能需要开启【电商增值】

1.绑定会员

构造会员信息对象KsEcv，且初始化时会员ID和会员名称为必填参数

构造回调SdkVipCallback

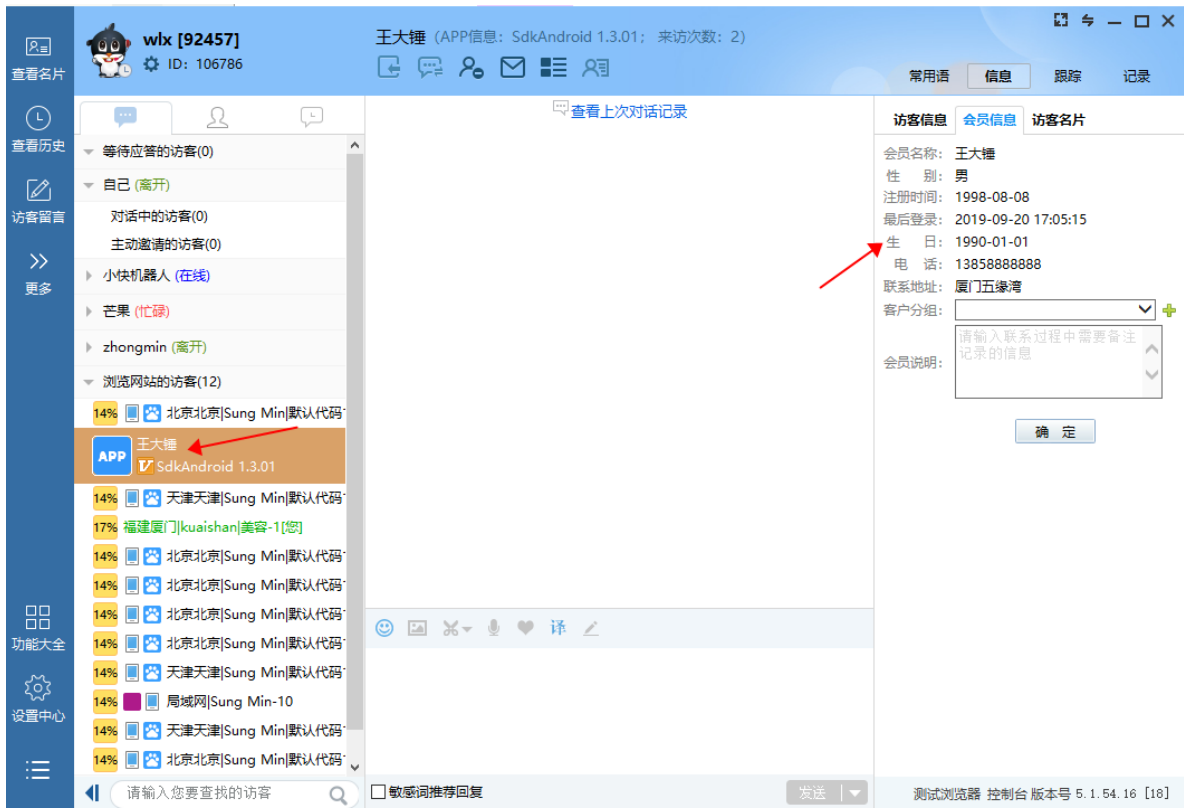
```

KSEcv ecv =new KSEcv("xxxxxx", "王大锤", "1998-08-08", "2019-09-01 09:00:00",
"男", "1990-01-01", "13858888888", "厦门五缘湾");
Sdkvipcallback callback = new Sdkvipcallback() {
    @Override
    public void onFail(String errorMessage) {
        //TODO 失败回调
    }

    @Override
    public void onResult(String result) {
        //TODO 成功回调
    }
};
KSConfig.bindVip(context, ecv,callback);

```

桌面客服端效果图



2.发送商品信息足迹

构造会员信息对象KsEcv, 会员ID和会员名称为必填参数

构造商品信息对象KsEcgc, 商品ID, 商品名称, 商品价格和商品图片为必填参数

构造回调SdkVipCallback

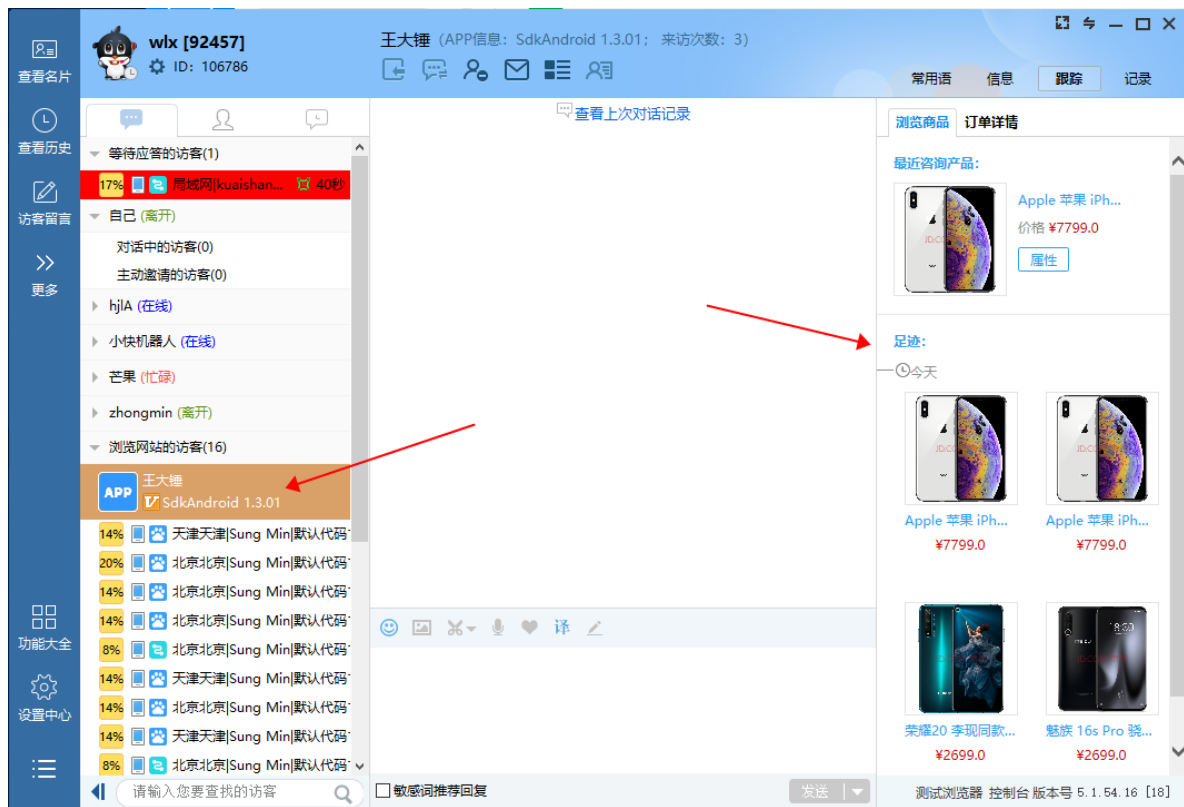
```

KsEcv ecv = new KsEcv("xxxxxx", "王大锤", "1998-08-08", "2019-09-01 09:00:00",
    "男", "1990-01-01", "13800000000", "厦门五缘湾");
Map<String, Object> attributes = new HashMap<>();
    attributes.put("厂商", "魅族");
    attributes.put("标签", "雄起");
    String attribute = new JSONObject(attributes).toString();
KsEcgc ecgc = new KsEcgc("meizu00001", "魅族 16s Pro 骁龙855plus 全屏屏拍照游戏手机
8GB+128GB 黑之谧镜 全网通移动联通电信4G手机 双卡双待",
2699, "https://img14.360buyimg.com/n0/jfs/t1/67351/34/8538/152014/5d6638e4E0d110c
21/42cd2c435ba01815.jpg", "https://item.jd.com/100004349203.html#crumb-wrap",
attribute);
sdkvipcallback callback = new sdkvipcallback() {
    @Override
    public void onFail(String errorMessage) {
        //TODO 失败回调
    }

    @Override
    public void onResult(String result) {
        //TODO 成功回调
    }
};
KsConfig.goodsTrack(context, ecv, ecgc, callback);

```

桌面客服端效果图



3.发送商品并打开咨询对话

构造会员信息对象KsEcv，会员ID和会员名称为必填参数

构造商品信息对象KsEcgc，商品ID，商品名称，商品价格和商品图片为必填参数

构造回调SdkVipCallback

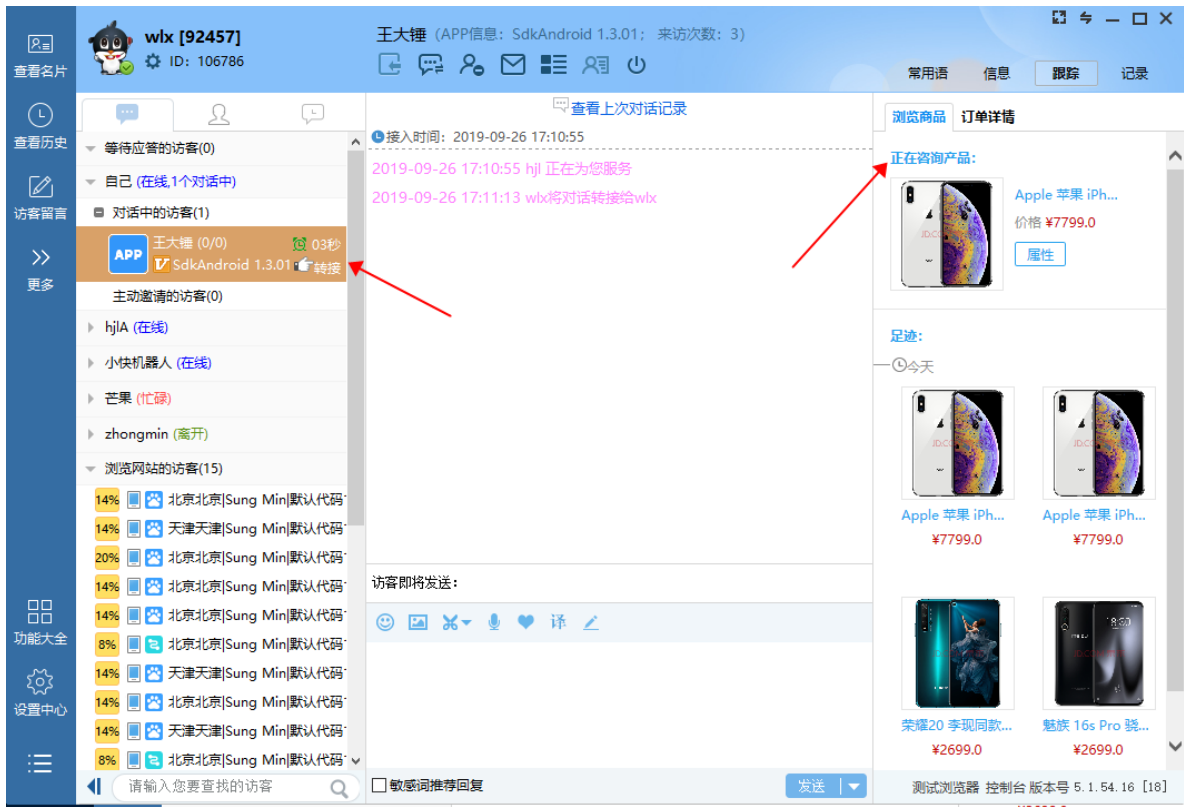
```

KsEcv ecv = new KsEcv("xxxxxx", "王大锤", "1998-08-08", "2019-09-01 09:00:00",
"男", "1990-01-01", "13800000000", "厦门五缘湾");
Map<String, Object> attributes = new HashMap<>();
    attributes.put("厂商", "魅族");
    attributes.put("标签", "雄起");
    String attribute = new JSONObject(attributes).toString();
KsEcgc ecgc = new KsEcgc("meizu00001", "魅族 16s Pro 骁龙855plus 全面屏拍照游戏手机
8GB+128GB 黑之谧镜 全网通移动联通电信4G手机 双卡双待",
2699, "https://img14.360buyimg.com/n0/jfs/t1/67351/34/8538/152014/5d6638e4E0d110c
21/42cd2c435ba01815.jpg", "https://item.jd.com/100004349203.html#crumb-wrap",
attribute);
sdkvipcallback callback = new sdkvipcallback() {
    @Override
    public void onFail(String errorMessage) {
        //TODO 失败回调
    }

    @Override
    public void onResult(String result) {
        //TODO 成功回调
    }
};
KsConfig.goodsConsult(context, ecv, ecgc, callback);

```

桌面客服端效果图



手机端效果



4、会员下单

构造会员信息对象KsEcv, 会员ID和会员名称为必填参数

构造订单信息对象KsEcg, 商品ID, 商品名称, 商品价格、商品图片和商品数量为必填参数

构造订单信息对象KsEco, 订单ID, 订单时间, 订单收货人、收货人地址, 收货人电话、订单状态, 商品列表为必填参数

构造回调SdkVipCallback

```
KsEcv ecv = new KsEcv("xxxxxx", "王大锤", "1998-08-08", "2019-09-01 09:00:00",
    "男", "1990-01-01", "13800000000", "厦门五缘湾");
    Map<String, Object> attributes = new HashMap<>();
    attributes.put("厂商", "魅族");
    attributes.put("标签", "雄起");
    String attribute = new JSONObject(attributes).toString();
```

```

KSEcg ecg = new KSEcg("meizu00001", "魅族 16s Pro 骁龙855plus 全面屏拍照游戏手机 8GB+128GB 黑之谧镜 全网通移动联通电信4G手机 双卡双待",
2699, "https://img14.360buyimg.com/n0/jfs/t1/67351/34/8538/152014/5d6638e4E0d110c21/42cd2c435ba01815.jpg", "https://item.jd.com/100004349203.html#crumb-wrap",
1,attribute);

KSEcg ecg2 =new KSEcg("meizu00002", "魅族 16th Plus 全面屏游戏手机 6GB+128GB 极光蓝 全网通移动联通电信4G手机 双卡双待",
1998, "https://img14.360buyimg.com/n0/jfs/t1/3090/8/6463/198259/5ba317d0E97f6ad48/62be44d2d99f40fa.jpg", "https://item.jd.com/100000504466.html#crumb-wrap",
2,attribute);

List<KSEcg> ecgList = new ArrayList<>();
ecgList.add(ecg);
ecgList.add(ecg2);
KSEco eco = new KSEco("order000001", "2019-10-23 10:00:00", "小明", "福建厦门集美", "1231111111", "银行汇款/转账", "2019-10-24 12:00:00", "顺丰快递", 1, "迅速点, 急用", ecgList, "tt_mb", 22222);
SdkvipCallback callback = new SdkvipCallback() {
@Override
public void onFail(String errorMessage) {
//TODO 失败回调
Log.v("MainActivity", errorMessage);
}

@Override
public void onResult(String result) {
Log.v("MainActivity", result);
//TODO 成功回调
}
};
ksconfig.placeOrder(context, ecv, eco, callback);

```



5、修改订单状态

构造会员信息对象KsEcv, 会员ID和会员名称为必填参数

构造订单信息对象KsEcg, 商品ID, 商品名称, 商品价格、商品图片和商品数量为必填参数

构造订单信息对象KsEco, 订单ID和订单状态必填参数

构造回调SdkVipCallback

```
KSEcv ecv = new KSEcv("xxxxxx", "王大锤", "1998-08-08", "2019-09-01 09:00:00",
"男", "1990-01-01", "13800000000", "厦门五缘湾");
KSEco eco = new KSEco();
eco.setOrderNo("order000001");
eco.setStatus(1)
sdkvipcallback callback = new Sdkvipcallback() {
    @Override
    public void onFail(String errorMessage) {
        Log.v("MainActivity",errorMessage);
        //TODO 失败回调
    }

    @Override
    public void onResult(String result) {
        Log.v("MainActivity",result);
        //TODO 成功回调
    }
};

KSConfig.updateOrderStatus(context, ecv, eco, callback);
```



四、消息通知

动态注册广播接受器, 监听新到消息通知

广播接受器

```
class MyReceiver extends BroadcastReceiver
{
    @Override
    public void onReceive(Context context, Intent intent) {
```



```
        List<Map<String,String>> list = (List<Map<String, String>>)
intent.getSerializableExtra("data");
        for (Map<String,String> item:list){
            String addTime = item.get("addTime");
            String content = item.get("content");
        }
        //系统提示音
        Uri uri =
RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);
        Ringtone rt = RingtoneManager.getRingtone(getApplicationContext(),
uri);

        rt.play();

    }
}
```

动态注册广播

```
myReceiver = new MyReceiver();
intentFilter = new IntentFilter();
intentFilter.addAction(KSConstant.ACTION_MESSAGE_NOTIFICATION);

LocalBroadcastManager.getInstance(this).registerReceiver(myReceiver,intentFilter);
```