

Worldpay Android Integration

Quick Start

Include the following permissions in AndroidManifest.xml

```
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS"/>
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
<uses-permission android:name="android.permission.MOUNT_UNMOUNT_FILESYSTEMS"
    tools:ignore="ProtectedPermissions" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.BLUETOOTH"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
<uses-feature android:name="android.hardware.usb.host" />
```

Import Universal_SDK_1.00.166.jar (or later) into the project.

Have the class implement OnReceiverListener, OnReceiverListenerPIN, WorldpayListener

```
public class MainActivity extends AppCompatActivity implements OnReceiverListener,
WorldpayListener, OnReceiverListenerPIN
```

Create a new IDT_Device instance

```
device = new IDT_Device(thisActivity,thisActivity);
```

Connect to the target device and register receiver. This example will connect to a VP3300 on USB (please refer to complete SDK documentation at ID Tech Products Homepage for examples how to connect to various devices).

```
device.device_setDeviceType(ReaderInfo.DEVICE_TYPE.DEVICE_VP3300_USB);
device.registerListen();
```

Monitor the Connect/Disconnect callback to understand device connection status

```
@Override
public void deviceConnected() {
    isReaderConnected = true;
    handler.post(doUpdateLabel);
}

@Override
public void deviceDisconnected() {
    isReaderConnected = false;
    handler.post(doUpdateLabel);
}
```

Create a WorldPay data object and populate it with transaction details

```
if (wp == null) wp = new WorldPay();

WorldPay.WorldPayData data = new WorldPay.WorldPayData();

data.accountID = "1188346";
data.accountToken =
"E1EB3EFB049DFB599F1CB454E1CFC4FD14BF90BCE74456AE9E9490D7D609B466C81A3801";
data.acceptorID = "364798674";
data.license = "2y2r0h6sorxb-5ufv0z2ao3jb"; //if live/production transaction, a valid
license must be provided
data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_SALE; //specify Sale,
Pre-Auth, or Completion
// data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_PRE_AUTH;
//specify Sale, Pre-Auth, or Completion
// data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_COMPLETION;
//specify Sale, Pre-Auth, or Completion
// data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_VOID;
// data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_RETURN;
// data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_QUICK_CHIP_START;
Start QuickChip -> Collect Card Data
// data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_QUICK_CHIP_FINISH;
Finish QuickChip -> Ready to submit with final amount known
data.amount = "1.00"; //specify transaction Amount
data.timeout = 30; //Timeout for transaction
data.isTest = true; //if test transaction, set to TRUE. If live transactions, set
to FALSE
data.msrOnly = false; //for EMV + MSR, set to FALSE. If MSR swipe only, set to TRUE
data.transactionID = ""; //required if COMPLETION. Otherwise, leave blank
data.referenceNumber = ""; //required value. If blank, SDK will put in random number
data.terminalID = "12345"; //required value. Set a terminal identifier
data.ticketNumber = ""; //required value. If blank, SDK will put in random number
data.duplicateCheck = false; //set to TRUE if you want to check for duplicate
transactions
data.duplicateOverride = true; //set to TRUE if you want to allow duplicate charges
to same card
data.enableCTLS = true; //set to TRUE if you want to allow Contactless transactions
(VP3300/VP8300 only)
data.declinePartial = true; //set to TRUE if you want not allow partial approval
from completing

wp.returnRequest = true; //set to TRUE if you would like the request packet returned
with the response
```

Execute the transaction

```
boolean requestOnly = false; // If this is TRUE, the SDK will return the request
packet instead of contacting WorldPay with it. For integrations that want to control
communication with WorldPay servers

int ret = wp.executeTransaction(device,data,thisActivity,requestOnly);
if (ret == ErrorCode.SUCCESS) {
    info += "Transaction Started Successfully";
    detail = "";
    handler.post(doUpdateStatus);
}
else {
```

```

    info += "Start Transaction Failed\n";
    info += "Status: "+ device.device_getResponseCodeString(ret)+"";
    detail = "";
    handler.post(doUpdateStatus);
}

```

Await and parse the results in the WorldPayCallback or WPCallback. The WorldPayCallback is the original callback, and the WPCallback is the updated callback that also returns receipt data and the original card data. You can use the receipt data to create a receipt, or use the original card data if any data elements are needed for review for investigation purposes. Please note, Android requires you implement BOTH callbacks, but you must only use ONE of them so you don't get duplicate responses back.

```

@Override
public void WorldPayCallback(String xmlResponse, int resultCode, int emvCode) {
    //While you must implement both WorldPayCallback or WPCallback,
    //only execute one callback, not both. In this example, we are not adding
    //any executable code to WorldPayCallback, and instead putting the code in
    //WPCallback below

    /*
    info += "WorldPay Callback\n";
    info += "Status Result Code: "+ WorldPay.getErrorString(resultCode)+"\n";
    info += "Status EMV Code: "+ String.valueOf(emvCode)+"\n";
    info += xmlResponse;

    detail = "";
    handler.post(doUpdateStatus);
    */
}

```

```

@Override
public void WPCallback(String xmlResponse, int resultCode, int emvCode, IDTEMVData
idtemvData, WorldPay.ReceiptData receiptData) {

    info += "WorldPay Callback\n";
    info += "Status Result Code: "+ WorldPay.getErrorString(resultCode)+"\n";
    info += "Status EMV Code: "+ String.valueOf(emvCode)+"\n";
    info += xmlResponse;
    if (receiptData != null){
        info += "\n\n";
        info += getReceipt(receiptData);
    }
    detail = "";
    handler.post(doUpdateStatus);
}

```

Reference page:

<https://atlassian.idtechproducts.com/confluence/display/KB/WorldPay+Integration+-+Home>

If your implementation is NOT using the IDT_Device class, but instead using a device specific class, you can send the sdkInstance as the first parameter for the execute transaction. Example:

Instead of this:

Create a new IDT_Device instance

```
device = new IDT_Device(thisActivity, thisActivity);  
int ret = wp.executeTransaction(device, data, thisActivity, requestOnly);
```

You can do this:

Create a new IDT_VP3300 instance

```
device = new IDT_VP3300(thisActivity, thisActivity);  
IDT_Device instance = IDT_VP3300.getSDKInstance()  
int ret = wp.executeTransaction(instance, data, thisActivity, requestOnly);
```

To execute successful CONTACT test transactions, you can use any UAT USA EMV card with Contact interface, or the Fiserv Dual Interface EMV Test Cards (2 card set). To execute successful CONTACTLESS test transactions, you can use UAT USA EMV Test Cards with Contactless interface, or the Fiserv Dual Interface EMV Test Cards (2 card set). Other test cards may work if they share the same account number as the defined cards. Do not use live cards in a test environment.

The device must be configured with proper terminal settings, AID's, and CAPKs. Please load the configuration file Worldpay_Device_Config.json to your device before running transactions. You can use SDK Demo, or UpdateIDT to perform the update

The device must have the correct key. If a test device, it must have demo BDK 0123456789ABCDEFFEDCBA9876543210, and if production, it must have a secure data encryption key (ID Tech Key IDT-KEYINJ-D04)

This SDK is for driving ID Tech devices to capture payment information and get approval/decline from WorldPay, and Void/Return. Other transaction types (Add Tip, etc) require the integrator to create their own request packets and send to WorldPay through their own methods. Please refer to Express API Interface Specification V3 for other request packets that can be created.

Sale vs QuickChip Transaction Types

```
data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_SALE
```

You use this transaction type if you know the final amount when the card is presented to the device. This will collect the card data and submit for approval at the amount you specify

```
data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_QUICK_CHIP_START
```

You use this transaction type if you do not know the final amount when the card is presented to the device. This will collect the card data and await the final transaction amount to be presented. Once the card data is collected, it does not need to be presented again after the final amount is known.

```
data.transactionType = WorldPay.TRANSACTION_TYPE.TRANSACTION_TYPE_QUICK_CHIP_FINISH
```

You use this transaction type once you know the final amount and the card data has already been previously collected with TRANSACTION_TYPE_QUICK_CHIP_START. If you attempt to execute this transaction type without the data previously collected, it will return an error. This will submit for approval at the final amount you specify without requiring the original card be presented again.

Going from Demo Mode to Production Mode

Once you are successful in performing test transactions, to perform live transactions, you will need to change the following four fields:

```
data.accountToken = ""; //needs production value
data.acceptorID = ""; //needs production value
data.isTest = false; //must be false to perform live transactions
data.license = ""; //needs production license
```

You will be provided production values for accountID, accountToken, acceptorID and license. In addition, you must change isTest to false so your transaction will be directed to the live production server at WorldPay.

Once you perform transaction, you can review them online by logging into your account at www.accessmyiq.com