ANDROID SDK
V3.2.3

Contents

Introduction 3
Supported Channels 3
Prerequisites 3
Download the SDK 4
The Flow 4
Integrating the SDK 5
Class and Interface 6
Initialization 7
Callback 8
Executing Payment 11
Other Important Methods 12
SDK Build 13
Queries 13

Introduction

Mimopay Android SDK is designed to help you, as a developer, or your developer team to integrate your android application or games to Mimopay's payment service. Mimopay Android SDK is equipped with an easy-to-understand User Interface and also easy to integrate.

Supported Channels

Topup:
1. Smartfren
2. Sevelin

ATM:
1. BCA
2. Bersama

Telkomsel Upoint:
1. Airtime
2. HRN (Voucher)

XL:
1. Airtime
2. HRN (Voucher)

Maxis:
1. Airtime

Digi:
1. Airtime

Celcom:
1. Airtime

Vietnam Telco:
1. VinaPhone
2. MobiFone
3. Viettel

Indosat:
1. Airtime

Prerequisites

Before you begin, you need to make sure that you are a registerred merchant at Mimopay payment gateway since the SDK will be useless without user ID, merchant code, and secret key that required to pass at initialization

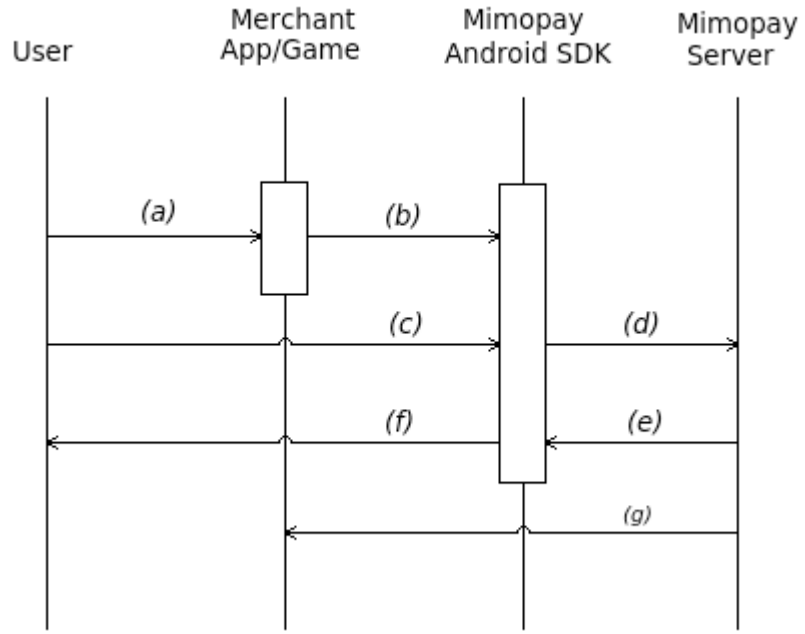
Download the SDK

The SDK is available at https://github.com/mimopay/Mimopay-Android-SDK. You can clone it with whatever git tool you have or use command line as shown below

```
git clone https://github.com/mimopay/Mimopay-Android-SDK.git
```

The Flow

UI Mode Flow Illustration



Where:
(a) User choose to do some payment.
(b) Merchant's app to activate SDK's built-in UI.
In this stage, merchant's app is become inactive. All interaction only between user and the SDK. The SDK provides necessary UI controls to simplify user to do the payment
(c) User choose some payment channel and method
(d) The SDK's UI pass all necessary things that user has choosen or input, to mimopay server
(e) Mimopay server replies back the results
(f) The SDK display the results
(g) Mimopay server than apply callback to merchant's server

Integrating the SDK

Before you start, you need to follow these steps below

1. You need to copy the Mimopay.jar into libs directory of your project.
2. The SDK require several android permissions so you need to add lines of codes below into the AndroidManifest.xml file of your project

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.READ_SMS" />
```

3. You need to add several line of codes below into the AndroidManifest.xml file of your project

```
<activity
        android:name="com.mimopay.MimopayActivity"
        android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen"
        android:windowSoftInputMode="stateUnspecified|adjustPan"
        >
</activity>
```

4. You also need to add these three line of codes into your java source.

```
import com.mimopay.Mimopay;
import com.mimopay.MimopayInterface;
import com.mimopay.merchant.Merchant;
```

Class and Interface

The following is the description of the SDK class and interface that you have imported in your java source code

```
public class Mimopay {
        public Mimopay(Context context,
                String emailOrUserId, String merchantCode, String productName,
                String transactionId, String secretKey, String currency,
                MimopayInterface mimopayinterface
        )
        // Topup
        public void executeTopup(String channel)
        // ATMs
        public void executeATMs(String channel)
        // Telkomsel Upoint
        public void executeUPointHrn()
        public void executeUPointAirtime()
        public void executeUPointAirtime(String amount)
        // XL
        public void executeXLAirtime()
        public void executeXLHrn()
        public void executeXLAirtime(String amount)
        // Maxis
        public void executeMPointAirtime()
        public void executeMPointAirtime(String amount)
        // Digi
        public void executeDPointAirtime()
        public void executeDPointAirtime(String amount)
        // Celcom
        public void executeCelcomAirtime()
        public void executeCelcomAirtime(String amount)
        // Vietnam Telco
        public void executeVnTelco()
        // Indosat
        public void executeIndosatAirtime()
        public void executeIndosatAirtime(String amount)
        // General functions
        public String getSdkVersion()
        public void enableGateway(boolean enable)
        public void enableLog(boolean enable)
        public String[] getLastResult()
        public void setCurrency(String currency)
}

public interface MimopayInterface {
        public void onReturn(String info, ArrayList<String> params);
}
```

Initialization

The SDK initialization require you to fill all your merchant stuffs into its parameters. Please refer to http://staging.mimopay.com/api.

```
Mimopay mimopay = new Mimopay(
        Context context,                    // your android application context
        String emailOrUserId,
        String merchantCode,
        String productName,
        String transactionId,
        String secretKeyStaging,
        String secretKeyGateway,
        String currency,
        MimopayInterface mi                 // your callback interface
);
```

transactionId may set to empty (transactioID = "") which cause the library to generate a unique value based on unix timestamp.

In the sample source codes, the secretKeyStaging and secretKeyGateway is managed as string of encrypted keys.

```
try {
    secretKeyStaging = Merchant.get(true, "zLdLLbLX7xi2E4zxcbGMPg==");
    secretKeyGateway = Merchant.get(false, "5aSkczdhkk4ukFsZEHykkA==");
} catch(Exception e) {
}
```

The encrypted values are bundled in a java JAR file that should be placed in the lib directory along with the SDK jar to make the final build. This will allow you to avoid your secretKey written in your java source codes.

Please contact us and we will generate for you a text file (encrypted keys) and JAR file (encrypted values).

Callback

To obtain any information from the SDK during and after execution, you need to use MimopayInterface interface and override its oReturn method as decribed below.

```
MimopayInterface mimopayinterface = new MimopayInterface()
{
        public void onReturn(String info, ArrayList<String> params)
        {
                if(info.equals("SUCCESS")) {
                        // do what you want to do
                } else if(info.equals("ERROR")) {
                        // do what you want to do
                } else if(info.equals("FATALERROR")) {
                        // do what you want to do
                }
        }
};
```

Use ‘info’ as indicator whether it is success, error, or fatal error status. Use ‘params’ to obtain the details. Be noticed that ‘params’ may set to **null** so you need to test first whether it equal to null or not. After that, you should obtain its size. Here is the ilustration

```
if(params != null) {
        int j = params.size();
        // ....
}
```

The following table visualize onReturn's info and params string

Info	Params		Meaning
ERROR	0	UserCancelled	User was cancelling the transaction
	0	MerchantLoadUrlNull	You pass the channel string other than listed in this document
	0	ErrorConnectingToMimopayServer	Internet connection problem
	0	ErrorValidatingVoucherCode	Occurs when SDK receive other then 'ok' status from server in during transaction process
	0	ErrorInternetConnectionProblem	Internet connection problem
	0	UnsupportedPaymentMethod	Requested payment method does not exist
	0	UnspecifiedChannelRequest	Requested payment channel does not exist
	0	ErrorHTTP404NotFound	Server return 404 error code.
	0	ErrorUnderMaintenance	The current payment channel is under maintenance
	0	ErrorInvalidPhoneNumber	Phone number that you've supplied is in invalid format
	0	ErrorInvalidDenomValue	The denom value that you've supplied is in invalid format
	0	ErrorInvalidAmountValue	The amount value that you've supplied is in invalid format
	Notes: Those error listed above are errors that occurs during transaction process. Since all transactions are done by Mimopay's server and also depend on the input, at the end of transaction, server might return an error or failed. Therefore since it is considerred as the end of the transaction (not 'during' transaction), then SDK will always return SUCCESS once it received status=200. ***In other word, the transaction success but the result is failed/error***.		
SUCCESS	0..3	Depends on the channel	Message retrieved from server
FATALERROR	0	Depends on the uncaughtException message	Occurs when the SDK suddenly force closed by the Android OS

Payment Method	info		params			
		0	1	2	3	4
Smartfren	SUCCESS	smartfren	OK	Sukses		PSS
		smartfren	FAILED	*SF1		PSS
	ERROR	LE	*HST	*JEM		PSS
Sevelin	SUCCESS	sevelin	OK	Sukses		PSS
		sevelin	FAILED	*SV1		PSS
	ERROR	LE	*HST	*JEM		PSS
ATM BCA	SUCCESS	atm_bca	*AT1	*AT2	*AT3	PSS
	ERROR	LE	*HST	*JEM		PSS
ATM Bersama	SUCCESS	atm_bersama	*AT1	*AT2	*AT3	PSS
	ERROR	LE	*HST	*JEM		PSS
Upoint HRN (voucher)	SUCCESS	upoint_hrn	OK	Sukses		PSS
		Upoint_hrn	FAILED	*SF1		PSS
	ERROR	LE	*HST	*JEM		PSS
Upoint Airtime	SUCCESS	upoint_airtime	*UP1	*AT1	*AT2	PSS
	ERROR	LE	*HST	*JEM		PSS
	ERROR	LE	*HST	*JEM		PSS
XL Airtime	SUCCESS	xl_airtime	*XL1	*AT1	*AT2	PSS
	ERROR	LE	*HST	*JEM		PSS
	ERROR	LE	*HST	*JEM		PSS
XL HRN (voucher)	SUCCESS	xl_hrn	OK	Sukses		PSS
		xl_hrn	FAILED	*XV1		PSS
	ERROR	LE	*HST	*JEM		PSS
Maxis Airtime	SUCCESS	mpoint_airtime	OK	*AT2		PSS
	ERROR	LE	*HST	*JEM		PSS
	ERROR	LE	*HST	*JEM		PSS
Digi Dpoint	SUCCESS	dpoint_airtime_ charge	OK	*AT2		PSS
	ERROR	LE	*HST	*JEM		PSS
	ERROR	LE	*HST	*JEM		PSS
Celcom	SUCCESS	celcom_airtime	OK	*AT2		PSS
	ERROR	LE	*HST	*JEM		PSS
	ERROR	LE	*HST	*JEM		PSS
VnTelco	SUCCESS	topup_vn	OK	Success		PSS
		topup_vn	FAILED	*VT1		PSS
	ERROR	LE	*HST	*JEM		PSS
Indosat Airtime	SUCCESS	indosat_airtime	*IS1	*AT1	*AT2	PSS
	ERROR	LE	*HST	*JEM		PSS
	ERROR	LE	*HST	*JEM		PSS

*Empty cell on params column means equals to **null***
LE = Listed Errors. Please refer to previous table
** = Please refer to table below*
*PSS = PaymentSubmitStage**X***

*X is indicates the number of payment submitted to mimopay backend server. For example, if params[4] equals to **PaymentSubmitStage_0**, that's means the current payment process has not been proceeded and recorded on mimopay backend server.*
X = 1,2,..n

SF1	Kode voucher sudah digunakan (voucher code has been used already)
VT1	Failed
SV1	Kode Voucher salah (wrong voucher code)
AT1	Company Code
AT2	Total Amount
AT3	Transaction ID
UP1	6-digit UP Code (upoint)
AT1	Destination Number for the for sending the shortcode
AT2	User phone number
XL1	4-digit XL's shortcode
XV1	Reload Gagal
MP1	You will receive a sms soon, please follow the instruction in it to complete this payment
DG1	SMS CODE is comming in
HST	Http status code or Java Exception (-1)
JEM	Java Exception status
IS1	Keyword BELI followed by 3 digit shortcode

Executing Payment

The following table will help you to understand the SDK's methods and how they are categorized.

Pymt Method	Channel	Built-in UI
Topup	smartfren sevelin	• void executeTopup(String channel)
ATM	atm_bca atm_bersama	• void executeATMs(String channel) • void executeATMs (String channel, String amount)
Telkomsel	upoint upoint_hrn	• void executeUPointHrn() • void executeUPointAirtime() • void executeUPointAirtime(String amount)
XL	xl_airtime xl_hrn	• void executeXLAirtime() • void executeXLAirtime(String amount) • void executeXLHrn()
Maxis	mpoint	• void executeMPointAirtime() • void executeMPointAirtime(String amount)
Digi	dpoint	• void executeDPointAirtime() • void executeDPointAirtime(String amount)
Celcom	celcom	• void executeCelcomAirtime() • void executeCelcomAirtime(String amount)
Vietnam Telco	vnp (VinaPhone) vms (MobiFone) vte (Viettel)	• void executeVnTelco()
Indosat	Indosat_airtime	• void executeIndosatAirtime() • void executeIndosatAirtime(String amount)

For Airtime payment methods, the SDK has its ability to allow user to send telco's shortcode SMS automatically however, it still need user intervention by showing a dialog message window to allow user to type the shortcode.
Parameter explanation:

String channel
channel should be one of the followings
smartfren
sevelin
atm_bca
atm_bersama
upoint
upoint_hrn
xl_airtime
xl_hrn
mpoint
dpoint
celcom
topup_vn
indosat_airtime

String code
A voucher code string for smartfren, sevelin, xl_hrn,and upoint_hrn topup

String amount
A mimocard value for atm_bca & atm_bersama

Other Important Methods

- For some reason you may want to know the current SDK version. You may call getSdkVersion() to obtain the current SDK version
- By default, the SDK points to staging.mimopay.com during execution. On your production release, you should use gateway.mimopay.com instead of staging. To enable it you call enableGateway(boolean enable) with enable sets to true
- The last successful transaction always be recorded. You can call getLastResult() to obtain those transaction messages.
- Sometimes we need to visualize SDK's log activities during process. You may enable it by calling enableLog(boolean enable) with enable sets to true
- Since version v2.2 SDK's built-in UI is now support custom language. Please refer to CustomLang.java source code, it shows you how to manage all words of your desired language in a single array of strings. After that then you need to call setUiLanguage(String[] slang)

SDK Build

The SDK builds with Android API Level 21

Queries

If you have any queries or issues please email me at jimmy@mimopay.com or skype to jimmybas