

GPay SDK integration for Android

Revision : 1.2

Last Changed : 16-Jan-2018

GPay SDK integration for Android : https://github.com/gmobi-github/gpay_sdk

Integration:

- Import gpay_sdk-19.0.0-release.aar as a module.
- Make sure that app has a module dependency on gpay_sdk-19.0.0-release.aar.
- Require minSdkVersion 17.
- GPay will be needing RECEIVE_SMS, READ_SMS, SEND_SMS, READ_PHONE_STATE permissions handled within the SDK.
- **Paytm, Go2Pay Wallet and Jio Money** are enabled on staging server.
- **Pay via Operator (WAP) and mPesa** are enabled on production server.
- **Pay via Operator (SMS)** is enabled for either case.

Staging Credentials:

- Partner Code : 5151
- Secret Key : test12345
- Use following details to test the Paytm Wallet:
 - Mobile Number – 7777777777
 - Password – Paytm12345
 - OTP – 489871
 - After every 5 minutes, the Wallet balance is topped up to 7,000 Rs.

Live Credentials:

- Partner Code : 1450
- Secret Key : test12345

Payment Transaction Steps:

1. Obtain the *PayPGService* instance. Either obtain staging service or obtain production service depending on your need. *PayPGService.getStagingService()* will return the *PayPGService* object pointing to staging environment. *PayPGService.getProductionService()* will return the *PayPGService* object pointing to production environment.

```
PayPGService Service = null;
```

```
// Getting the PayPGService Instance. PayPGService.getStagingService() will return  
// the service pointing to staging environment.
```

```
Service = PayPGService.getStagingService();
```

```
//TODO use either of both as per requirement
```

```
// Getting the PayPGService Instance. PayPGService.getProductionService() will return  
// the service pointing to production environment.
```

```
Service = PayPGService.getProductionService();
```

2. Create a *PayReqParams* object holding the payment Information. Syntax for creating *PayReqParams* object is as follows:
If transaction amount equals 1,3,5,7,10,15,20,35,49 then only WAP/SMS billing option will be enabled else disabled.

```
// PayReqParams object creation
PayReqParams payReqParams = new PayReqParams();
payReqParams.setAppName("app name");    // Application name (mandatory)
payReqParams.setPtxid("TXN_12345678"); // Transaction Id (mandatory)
payReqParams.setAmount("10");           // Transaction amount as integer only (mandatory)
payReqParams.setImeiIp("1234567890123456"); // Device IMEI number (mandatory)
payReqParams.setPartnerCode("1234");    // Partner code provided by GMobi(mandatory)
payReqParams.setSecretKey("1234");       // Secret key provided by GMobi (mandatory)
payReqParams.setAttributes(attrs);       // Pay dialog attributes, for setting alert message for
PAY via Operator (SMS) (optional)
```

3. **(Optional)** Create *PayDialogAttrs* object holding the dialog attributes information. Syntax for creating *PayDialogAttrs* object is as follows:

```
PayDialogAttrs attrs = new PayDialogAttrs();
attrs.setDialogAlertMsg(null);
attrs.setHidePaytmWallet(false);
attrs.setHideAlertMsg(false);
attrs.setHideGo2PayWallet(false);
attrs.setHideJioMoney(false);
attrs.setHideMPesa(false);
attrs.setHideSMSBilling(false);
attrs.setHideWapBilling(false);
```

4. Call the *initialize()* method to set *PayReqParams* object. Method signature is as follows:

```
// Call this method and set PayReqParams object before starting transaction.
Service.initialize(payReqParams);
```

5. Call *startTransaction()* method to start the payment transaction. Method signature is as follows:

```
//Start the Payment Transaction. Before starting the transaction ensure that initialize method is
called.
Service.startTransaction(MainActivity.this, this);
```

6. Implement the callback methods to handle the response upon payment completion. Callback methods are under *PayTransactionCallback*

```

@Override
public void onTransactionResponse(PayResParams payResParams) {
    Log.e(TAG, "pay_response --> " + payResParams.toString());
}

@Override
public void onBackPressedCancelTransaction() {
    // TODO Auto-generated method stub
}

@Override
public void onTransactionCancel(String inErrorMessage) {
    Toast.makeText(MainActivity.this, inErrorMessage, Toast.LENGTH_SHORT).show();
}

```

PayDialogAttrs Request Parameters (Optional):

S. No.	PayDialogAttrs Request Parameter	Parameter type	Description
1	hideAlertMsg	Boolean (default false)	You can use setHideMessage() to show/hide payment alert message (Optional)
2	hidePaytmWallet	Boolean (default false)	You can use setHidePaytmWallet() to show/hide paytm wallet option (Optional)
3	hideJioMoney	Boolean (default false)	You can use setHideJioMoney() to show/hide jio money option (Optional)
4	hideMPesa	Boolean (default false)	You can use setHideMPesa() to show/hide mPesa option (Optional)
5	hideGo2PayWallet	Boolean (default false)	You can use setHideGo2PayWallet() to show/hide go2pay wallet option (Optional)
6	hideWapBilling	Boolean (default false)	You can use setHideWapBilling() to show/hide wap billing option (Optional)
7	hideSMSBilling	Boolean (default false)	You can use setHideSMSBilling() to show/hide sms billing option (Optional)
8	dialogAlertMsg	String	You can use setDialogAlertMsg() to set payment alert message (Optional)

PayReqParams Request Parameters:

S. No.	PayReqParams Request Parameter	Parameter type	Description
1	amount	String	You can use setAmount() to set the value of transaction amount (mandatory)
2	ptxid	String	You can use setPtxid() to set the value of payment transaction amount (should be integer value) (mandatory)
3	appName	String	You can use setAppName() to set the value of application name (mandatory)
4	imeiIp	String	You can use setImeiIp() to set the value of device imei (mandatory)
5	partnerCode	String	You can use setPartnerCode() to set the value of code provided regarding a particular partner by GMobi (mandatory)
6	attributes	Object of PayDialogAttrs	You can use setAttributes() to set the value of attributes (Optional)
7	secretKey	String	You can use setSecretKey() to set the value of secret key provided by GMobi (mandatory)

Response Parameters:

S. No.	Response Parameter	Parameter type	Description
1	status	String	You can use getStatus() to get the value of transaction status. Possible Values : cancelled, pending, success, failed.
2	message	String	You can use getMessage() to get the value of transaction message
3	amount	String	You can use getAmount() to get the value of transaction amount
4	ptxid	String	You can use getPtxid() to get the value of transaction id
5	txid	String	You can use getTxid() to get the value of transaction id provided by GMobi

ScreenShot :