

مصرف الراجحي
Al Rajhi Bank



ARB Payment Gateway SDK 1.0

Merchant Integration Guide

Version 1.1

Payment Gateway – Merchant Integration Guide – 1.3.

All Rights Reserved

These materials are confidential and proprietary to ARB and no part of these materials should be reproduced, published in any form by any means, electronic or mechanical including photocopy or any information storage or retrieval system nor should the materials be disclosed to third parties without the express written authorization of ARB.

Payment Gateway – Merchant Integration Guide – 1.0

Documentation Control

Ver. No.	Author/s	Reason for Change	Date of Change	Reviewed by	Reviewed Date
1.0	Al Rajhi Bank	Initial Draft	01-01-2020		
1.1	Al Rajhi Bank	MOF Details	16-12-2022		

Payment Gateway - Merchant Integration Guide - 1.0

Executive Summary

This document is prepared to provide information about integrating merchant application with Payment Gateway SDK.

Payment Gateway – Merchant Integration Guide – 1.0

About the Document

In order that the merchants are integrated in a secure and mandated manner, this reference document is being shared. The expectation being that the merchant's system integrator or auditor is able to refer to a document while performing integration as well as post integration. It contains the technical integration details including message formats to be used in communicating to the Payment Gateway SDK irrespective of the merchant platform being used. The document also shares the best practices and recommendations the merchant should follow during the integration with Payment Gateway SDK.

Table of Contents

Documentation Control	3
Executive Summary	4
About the Document	5
1. Merchant Prerequisites	7
2. Merchant Integration Process	8
2.1 To login into Payment Gateway application follow the below steps	8
2.2 To download Plugin to your system, follow the steps	11
2.3 Integrate the SDK.aar with merchant application	13
2.4 HMAC Generation	40

1.Merchant Prerequisites

1. Android Studio
2. Android SDK

2. Merchant Integration Process

Integration Process covers the process to be followed by Merchants.

Steps to be followed by Merchants:

- 1) Merchant should login through payment gateway URL using their Merchant ID, User ID and Password
URL: <<Live URL should be provided here>>
- 2) Merchant should download the Payment gateway SDK's ARR to connect Payment Gateway
- 3) Merchant should be informed about **Tranportal ID, Tranportal Password and Terminal Resource Key(HTTP Transaction)** which is generated for the specific Merchant and Terminal by the Bank
- 4) Construct the Request message as expected by Payment gateway SDK
- 5) Process the response message receiving the response from Payment gateway SDK

2.1 To login into Payment Gateway application follow the below steps

 Type the application URL (Live URL should be provided) in the browser and click Enter. Payment Gateway user login page is displayed as shown below:

Figure 1: Merchant Login Screen

The image shows a web browser window titled "Login". At the top left, there is a "Merchant" label with a small icon. Below this, there are four input fields stacked vertically, labeled "Institution ID", "Merchant ID", "User ID", and "Password". To the right of the "Institution ID" field is a "Merchant" label. Below the "Password" field is a Captcha section consisting of a red 'S' logo, a refresh icon, and a text input field labeled "Captcha". At the bottom of the form, there are three buttons: a red "Submit" button, a grey "Reset" button, and a link labeled "Forgot Password" with a key icon.

Please enter the information communicated to you through email (**Institution ID**, **Merchant ID**, **User ID**, **Password**) & **CAPTCHA** (dynamically generated text flashed on the screen)

Click **Submit**. The application displays the following "Home Page" screen with menus.

Figure 2: Home Page Screen

26 June 2019



Welcome merchant...



Logout



Dashboard

[Click here to download merchant user manual](#)

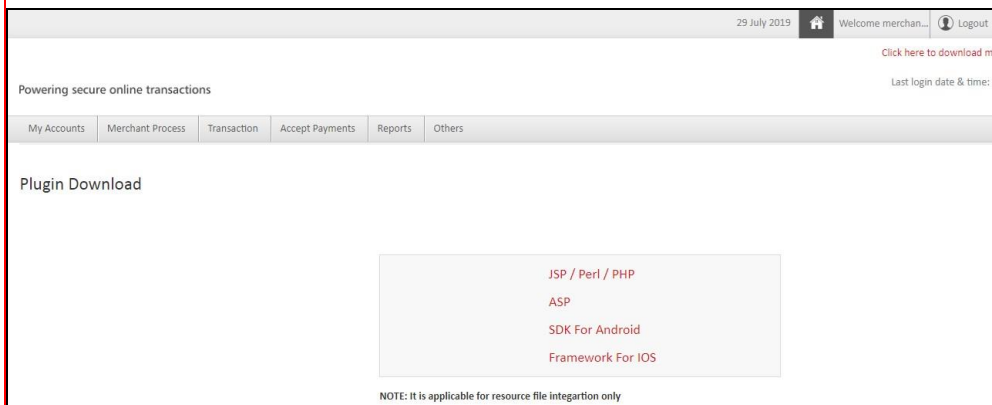
Last login date & time: 26-06-2019 15:51:38

My Accounts	Merchant Process	Transaction	Accept Payments	Reports	Others
-------------	------------------	-------------	-----------------	---------	--------

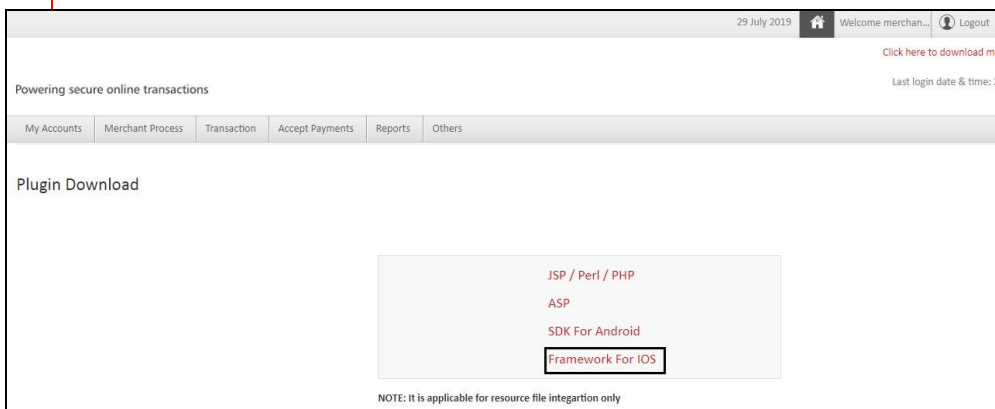
2.2 To download Plugin to your system, follow the steps

1. Click Merchant Process SDK Download; the application displays the following screen:

Figure 3: Plugin Download Screen



2. a) To download Java Plugin, click **SDK AAR** link; the application displays the 'File Download' dialog box:



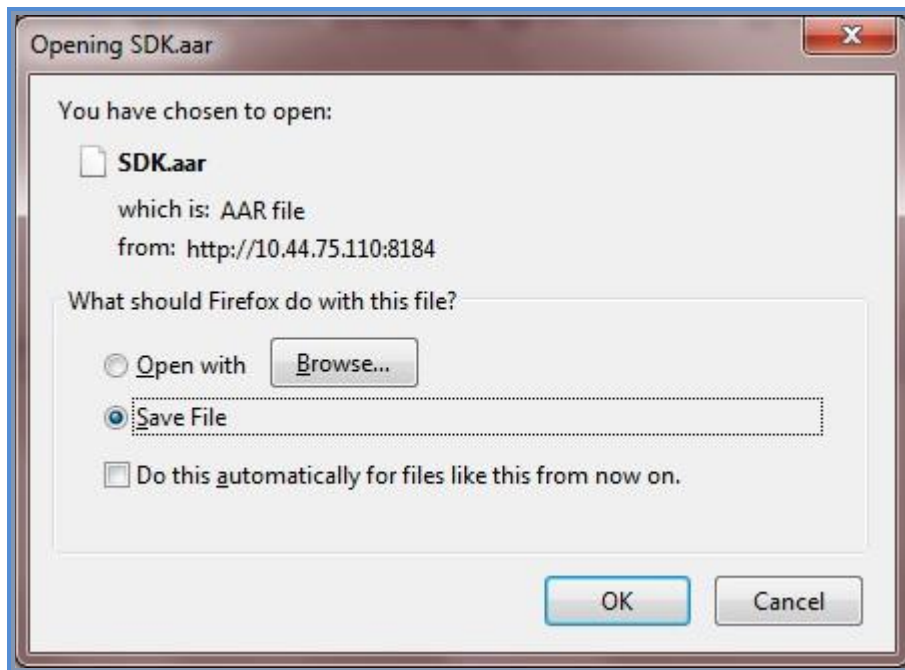


Figure 4: File Download Dialog Box

Note:

Merchant have to get the SDK.aar and have to do the following steps.

STEP 1: In Android Studio File -> New -> New Module ->

Import.jar/.aar and import SDK.aar

STEP 2: Then in your project's build.gradle (the one under 'app') add the following:

```
dependencies { implementation project('SDK')
}
```

STEP 3: Clean Build

2.3 Integrate the SDK.aar with merchant application

2.3.1 Merchant Request Parameter

The below are the details Merchant web application should send while connecting Payment Gateway

S. No	Merchant parameters to SDK	Description	Type	Max	Mandatory
1	id	Tranportal ID	Alpha Numeric	10	Yes
2	action	Transaction Action Type	Number	10	Yes
3	RESP_URL	Response URL	Text	255	Yes
4	ERROR_URL	Error URL	Text	255	Yes
5	amt	Amount	Number with decimals		Yes
6	currencycode	Currency Code	Number	3	Yes
7	trackid	Track ID [Text Field]	Numeric or Alpha Numeric	255	Yes
8	udf1	UDF 1 [Text Field - Optional]	Alpha Numeric	255	No ,Yes for ATM PIN transactions
9	Udf2	UDF 2 [Text Field - Optional]	Alpha Numeric	255	No ,Yes for

					ATM PIN transactions
10	Udf3	UDF 3 [Text Field - Optional]	Alpha Numeric	255	No
11	Udf4	UDF 4 [Text Field - Optional]	Alpha Numeric	255	No
12	Udf5	UDF 5 [Text Field - Optional]	Alpha Numeric	255	No
14	type	Card type	Text		
15	PUBKEY	Public key	Alpha Numeric		
16	TRAN_TYPE	HTTP, HTTP_VBV, TCPIP, TCPIP_VBV	Text		YES
17	TYPE	Hosted or Tranportal	Text		YES
18	ENC	Encryption to be used – AES or 3DES	Text		YES
19	HMAC	HMAC value of all the fields with PUBKEY as key	Alpha Numeric		YES
20	END_URL	Endpoint URL	Alpha Numeric		YES
21	card	Card Number	Numeric		YES for tranpo rtal transaction

22	expmonth	Expiry Month	Numeric	YES for tranpo rtal transac tion
23	expyear	Expiry Year	Numeric	YES for tranpo rtal transac tion
24	cvv	Card CVV	Numeric	YES for tranpo rtal transac tion
25	member	Card Holder Name	Text	YES for tranpo rtal transac tion
26	END_URL_TRAN VBV	PARES Endpoint URL	Alpha Numeric	YES for 3DSec ure Tranpo rtal TCPIP transac tion

27	payorIDType		Alphanumeric	<p>Payor ID Type</p> <p>Ex: NAT</p> <p>NAT – National ID</p> <p>IQA – IQAMA ID</p> <p>BIS – Business ID</p> <p>C700 – 700 Code</p> <p>UOI – Unified Organization ID</p> <p>GCC – GCC Passport Number</p> <p>PAS – Passport Number</p> <p>BDN – Border Number</p>
28	PayorIDNumber		Numeric	<p>Payor ID Number Ex: 1234567891</p> <ul style="list-style-type: none"> • For NAT, id number value should start with 1 and value length should be 10 digits • For IQA, id number value should start with 2 and value length should be 10 digits • For BIS, id number length should be 10 digits • For C700, id number length should be 10 digits • For the other id types, the id number should be between 6 and 15

29	billDetails	Agency Billing Details	String	1-20	Contains Bill details of the Agencies in JSON Array Format
----	-------------	------------------------	--------	------	--

Merchant App Bundle Parameters & Description:

1. **Tranportal ID:** Unique ID assigned for a Terminal under a merchant.

(ex: 101001)

2. **Transaction Action Type:** Describes the received transaction type

(ex: 1 – Purchase, 2 – Authorization, 17- IMPS etc)

3. **Response URL:** Merchant response URL

(ex: <http://10.44.71.46/demomerchant/successresult.jsp>); if authentication and authorization successful at the Payment Gateway then response would be sent to the Response URL.

4. **Error URL:** Merchant Error URL

(ex: <http://10.44.71.46/demomerchant/errorresult.jsp>); if authentication and authorization failed at the Payment Gateway then response would be sent to the error URL.

5. **Amount:** Transaction amount to be charged to the Customer for the product purchased (ex: Rs. 100.00).

6. **Currency:** ISO Currency code (ex: 356 [for INR]) to be set by the merchant for initiating the

7. **Track ID [Text Field]:** Allows you to map unique track ID to every transaction originated by the merchant (ex: 112312312313)

8. **UDF 1 [Text Field - Optional]:** Allows Merchant to send any additional information .In case of ATM PIN transaction merchant should send merchant response URL

9. **UDF 2 [Text Field - Optional]:** Allows Merchant to send any additional information .In case of ATM PIN Transactions merchant should send Merchant Error URL

10. **UDF 3 [Text Field - Optional]:** Allows Merchant to send any additional information

11. **UDF 4 [Text Field - Optional]:** Allows Merchant to send any additional information

12. **UDF 5 [Text Field - Optional]:** Allows Merchant to send any additional information
13. **Card type :** Determines which card type is being used for transaction (Ex : C or D)
14. **Public key:** Tranportal resource key shared by bank to merchant after generating specific key to each Terminal. (Ex:
91771655140891771655140891771655)

Note : Public key coresponse to Terminal resource key from terminal,Key Length may vary based on Encryption algorithm (AES or 3DES) that was used in Payment gateway.Key has to be shared to merchant before attempting transaction for HTTP type of transations.

15. **TRAN_TYPE :** Merchant has to send TRAN_TYPE based on which type of transaction it will support like TCPIP(Server to server call),HTTP(Browser redirection),HTTP_VBV (Redirection with 3DSecure),TCPIP_VBV (Server based with 3DSecure)
16. **TYPE :** Merchant should decide whether they will go for HOSTED(card detail captured in PG end) or TRANPORTAL (Card detail captured in merchant end) type of transaction
17. **ENC :** Determines which encryption has to be followed by payment Gateway with respect to particular bank . (Ex: AES).
18. **HMAC:** Merchant App and SDK can achieve integrity by commencing hashing between App and SDK.If hased value found mismatched transaction will not proceed further. (Ex: sdfseftrwetsjdejrtyiyouiipulul)
19. **END_URL:** Context URL of Payment Gateway to which transaction has to be initiated (Ex:

<https://products.ARB.co.in/iPayRMP/tranPipe.htm?param=tranInIt>)

20. **Card** : Customer Card Number is sent to SDK by merchant when merchant opt for tranportal transaction
21. **Expiry Month** : Customer Expiry month is sent to SDK by merchant when merchant opt for tranportal transaction
22. **Expiry Year** : Customer Expiry year is sent to SDK by merchant when merchant opt for tranportal transaction
23. **Card CVV** : Customer card cvv is sent to SDK by merchant when merchant opt for tranportal transaction
24. **Card holder name** : Card holder name is sent to SDK by merchant when merchant opt for tranportal transaction
25. **END_URL_TRANVBV** : For tranportal TCPIP transaction merchant has to send PARES received from ACS to SDK which in turn send it to payment gateway for verification.

2.3.2 Merchant Response Parameter :

The below are the details Payment Gateway SDK shall send to Merchant application.

S. No	Payment Gateway SDK Response Parameters	Type
1	Payment ID	Number
2	Result	Text
3	Reference ID	Number
4	Transaction ID	Number
5	Customer ID(In case of Standing Instruction)	Number
6	Post Date	Number
7	UDF 1 [Text Field - Optional]	Text
8	UDF 2 [Text Field - Optional]	Text
9	UDF 3 [Text Field - Optional]	Text
10	UDF 4 [Text Field - Optional]	Text
11	UDF 5 [Text Field - Optional]	Text

12	Amount	Number with decimals
13	Error	Text
14	Error Text	Text

Response Parameters Description:

1. **Payment ID:** Unique ID which is generated if the request from the merchant received and successfully validated at the payment gateway.
2. **Result:** Describes the result of the transaction (ex: **CAPTURED**).
3. **Reference ID :** RRN –generated by Base-24
4. **Transaction ID:** Unique ID generated by Payment Gateway for the transaction.
5. **Customer ID:** Unique ID generated by Payment Gateway for Standing Instruction.
6. **Post Date:** Transaction post date.
7. **UDF 1 [Text Field - Optional]:** Additional information
8. **UDF 2 [Text Field - Optional]:** Additional information
9. **UDF 3 [Text Field - Optional]:** Additional information
10. **UDF 4 [Text Field - Optional]:** Additional information
11. **UDF 5 [Text Field - Optional]:** Additional information
12. **Amount:** Transaction Amount

In case of failure transaction, Payment Gateway shall send the below mentioned parameters with Payment ID, Result, UDF fields, Amount.

1. **Error:** Describes the error with Error code.
2. **Error Text:** Describes the error with Error code.

Final Response and Description

S. No	Final Response	Description
1	CAPTURED or APPROVED or VOIDED	Result of successful transaction.
2	Error or ErrorText	Result of failure transaction.

2.3.3 Hosted Payment Integration

Merchant can connect SDK using below step

1. Call the SDK class as follows

```
new PGService().invokePGService(MainActivity.this, myBundle, new
PGService.Payment() {
@Override
public void onPaymentSuccess(Bundle resultBundle) {
//Handle Response
}
@Override
public void onPaymentFailure() {
}
});
```

2. Values to be sent in bundle

```
Bundle myBundle = new Bundle();
```

```
myBundle.putString("AMOUNT", "10.00");
myBundle.putString("ACTION", "1");
```

```
//Merchant Track ID
```

```
myBundle.putString("TRACK_ID", );
myBundle.putString("UDF1", "<Will be provided while onboard>");
myBundle.putString("UDF2", "<Will be provided while onboard>");
myBundle.putString("UDF3", "<Will be provided while onboard>");
myBundle.putString("UDF4", "<Will be provided while onboard>");
myBundle.putString("UDF5", "<Will be provided while onboard>");
myBundle.putString("UDF6", "<Will be provided while onboard>");
myBundle.putString("UDF7", "<Will be provided while onboard>");
myBundle.putString("UDF8", "<Will be provided while onboard>");
myBundle.putString("UDF9", "<Will be provided while onboard>");
myBundle.putString("UDF10", "<Will be provided while onboard>");
myBundle.putString("CURRENCY_CODE", "682");
myBundle.putString("LANGUAGE_ID", "EN");
myBundle.putString("PAYER_ID_TYPE", "NAT");
myBundle.putString("PAYER_ID_NUMBER", "1234567891");
```

```
//Terminal Resource key
```

```
myBundle.putString("TRAN_ID", "IPAYhhcM5l5WVY5");
```

```
//Terminal Resource password
```

```
myBundle.putString("TRAN_PASSWORD", "dJu71!0WgUL$3$b");
```

```
//Bill Details
```

```
myBundle.putString("BILL_DETAILS", [{"issuerAgencyId": "123456789012345678",
"billingAccountId": "54545454",
"billingCycle": "454544545454",
"dueAmount": "0.00",
"paidAmount": "10.00",
"billReferenceInfo": "8454545fdfsdfs",
"agencyCode": "MCI"}
]);
```

```

myBundle.putString("PUBKEY", "");
myBundle.putString("RESP_URL", "");
myBundle.putString("ERROR_URL", "");

```

BillDetails Description :

S.No	Fields	M/C/O	Field Type	Length	Occurrence	Owner
1	issuerAgencyId	M	String	18	1	Agency
2	billingAccountId	M	String	6-20	1	Agency
3	billingCycle	O	Nu- meric	1-12	1	Agency
4	dueAmount	M	Money		1	Agency
5	paidAmount	M	Money		1	Agency
6	billReferenceInfo	O	Char- acter	[Max 80]	1	Agency
7	agencyCode	M	String	4	1	Agency

Note – The Field Name should be same as mentioned in the formats, any difference (including spell mistake) would lead to error in processing the transaction.

1. Get the response from resultBundle

SUCCESS

```

{RESULT=CAPTURED,
CARD_TYPE=MASTERCARD,
AUTH_RESP_CODE=00,
TRAN_ID=202234890576464,
AVR=N,
REF=234861000305,
AUTH=831000,
UDF1=Test1,
UDF2=Test2,
UDF3=ARBPGMOF01,
UDF4=PGMOGTERM01,
UDF5=Test5,
UDF6=Test6,

```

UDF7=PG209200,
UDF8=PG209200,
UDF9=PG209200,
UDF10=PG209200,
TARCK_ID=891204,
POST_DATE=null,
AMOUNT=10.0,
PAYMENT_ID=600202234890624095}

FAILURE

[{ERROR_TEXT=!ERROR!-IPAY0300112-Unable to process the message,
RESULT=null,
CARD_TYPE=MASTERCARD,
AUTH_RESP_CODE=null,
TRAN_ID=202235066310328,
AVR=N, REF=235061000091,
AUTH=null,
UDF1=Test1,
UDF2=Test2,
UDF3=600000247,
UDF4=PG209200,
UDF5=Test5,
UDF6=Test6,
UDF7=PG209200,
UDF8=PG209200,
UDF9=PG209200,
ERROR=IPAY0300112,
UDF10=PG209200,
TARCK_ID=624951,
POST_DATE=null,
AMOUNT=10.0,
PAYMENT_ID=600202235033668938}]