






**API SPECIFICATIONS  
FOR  
HOSTED PAYMENT PAGE  
+  
3 ENDPOINTS  
+  
GENERAL OPERATIONAL FUNCTIONS  
  
INTEGRATION  
(Version 13.14)**

Last updated on 09th Jan 2020

	<b>RAZER MERCHANT SERVICES</b>
  	<p>J-39-1, Block J, Persiaran Multimedia, i-City, 40000 Shah Alam, Selangor, Malaysia.</p> <p>+(603) - 5521 8438</p> <p>support-sa@razer.com</p> <p>merchant.razer.com</p>
	<p><b>Social Networks</b></p> <p> <a href="https://twitter.com/Razer_MS">https://twitter.com/Razer_MS</a>  <a href="https://facebook.com/RazerMerchantServices">https://facebook.com/RazerMerchantServices</a>  <a href="https://youtube.com/c/RazerMerchantServices">https://youtube.com/c/RazerMerchantServices</a>  <a href="https://instagram.com/RazerMerchantServices">https://instagram.com/RazerMerchantServices</a>  <a href="https://linkedin.com/company/RazerMerchantServices">https://linkedin.com/company/RazerMerchantServices</a> </p>
	<p><b>Developer Platforms</b></p> <p> <a href="https://github.com/RazerMerchantServices">https://github.com/RazerMerchantServices</a>            Mobile XDK, seamless and inpage checkout, and many shopping carts            payment plugin/module/addon/extension are available         </p>

# ChangeLogs

2018/05/31	v13.8	Adding Pick-n-Pay APIs
2018/08/20	v13.9	Adding card BIN info API
2019/01/04	v13.9	Update value of channel in return URL, withdraw Pay-n-Pick from this document
2019/04/30	v13.9	Rebranding to Razer Merchant Services
2019/05/07	v13.10	Withdrawal of escrow and mass payout APIs
2019/06/17	v13.11	Adding currency & error code/desc to requery APIs response
2019/09/20	v13.12	Transferring predefined bank and FTT list to Razer Escrow & Mass Payout API
2019/11/29	v13.13	Adding Taiwan payment channels, removing EOL channels table
2020/01/09	v13.14	Adding eWallet channel (GrabPay, Maybank QRPay), 51 currencies supported for multi-currency card acceptance

# Table of Contents

[INTRODUCTION](#)

[PAYMENT FLOW OVERVIEW](#)

[SECURITY & DATA INTEGRITY](#)

[Verify Key \(Public Key\)](#)

[Secret Key \(Private Key\)](#)

[vcode \(in payment request\)](#)

[skey \(in payment response\)](#)

[SIMULATION](#)

[PAYMENT APIs](#)

[Initiating Payment](#)

[Channel Lists](#)

[Multi-Currency Payment](#)

[Malaysia](#)

[Singapore](#)

[Indonesia](#)

[China](#)

[Thailand](#)

[Vietnam](#)

[Philippines](#)

[Taiwan](#)

[Payment Page Integration](#)

[Getting Payment Result](#)

[Payment Status Notification \(Merchant Webhook or the 3 Endpoints\)](#)

[Return URL with IPN \(Instant Payment Notification\)](#)

[Notification URL with IPN](#)

[Callback URL with IPN](#)

[TL;DR?](#)

[SUMMARY ON INTEGRATION, 3 ENDPOINTS AND IPN](#)

[INTEGRATION](#)

[3 ENDPOINTS for payment response notification](#)

[IPN](#)

[Comparison Chart](#)

## MERCHANT REQUEST APIs

[Direct Status Requery](#)

[Indirect Status Requery](#)

[Daily Transaction Report \(Reconciliation\)](#)

[Settlement Report \(Reconciliation\)](#)

[Unsettled/Refunded Transaction Report \(Exclusion from settlement\)](#)

[Capture Request \(For pre-auth or authorized transaction\)](#)

[Reversal Request](#)

[Reversal Reference Table](#)

[Advanced Full/Partial Refund](#)

[Partial Refund Status Inquiry by TxnID/RefID](#)

[Appendix for Partial Refund and Partial Refund Status Inquiry](#)

[Static QR-Code Generator](#)

[Channel Status API](#)

[Card BIN information API](#)

[Foreign Exchange Rate API](#)

[Void Pending-Cash API](#)

[Recurring Plans API](#)

## ERROR CODES

[Payment Page](#)

[3DS Error](#)

[Payment Status Query \(PSQ\) Error](#)

[Error in Merchant Admin](#)

[Error Codes in Channel](#)

## RESOURCES

[Logos of all brand name](#)

[Mobile SDK/XDK](#)

[Supported Shopping Cart](#)

[ISO References](#)

[Handling JSON/PLAIN TEXT using .NET](#)

# INTRODUCTION

Dealing with banks and multiple payment channels are time-consuming and it might cost you a lot of money. We also understand that it's a hassle for those non-technical people to integrate even one kind of online payment method into their existing website. Thus we consolidate all payment channels into one integration to reduce merchant's costs and effort, and to shorten your time-to-market.

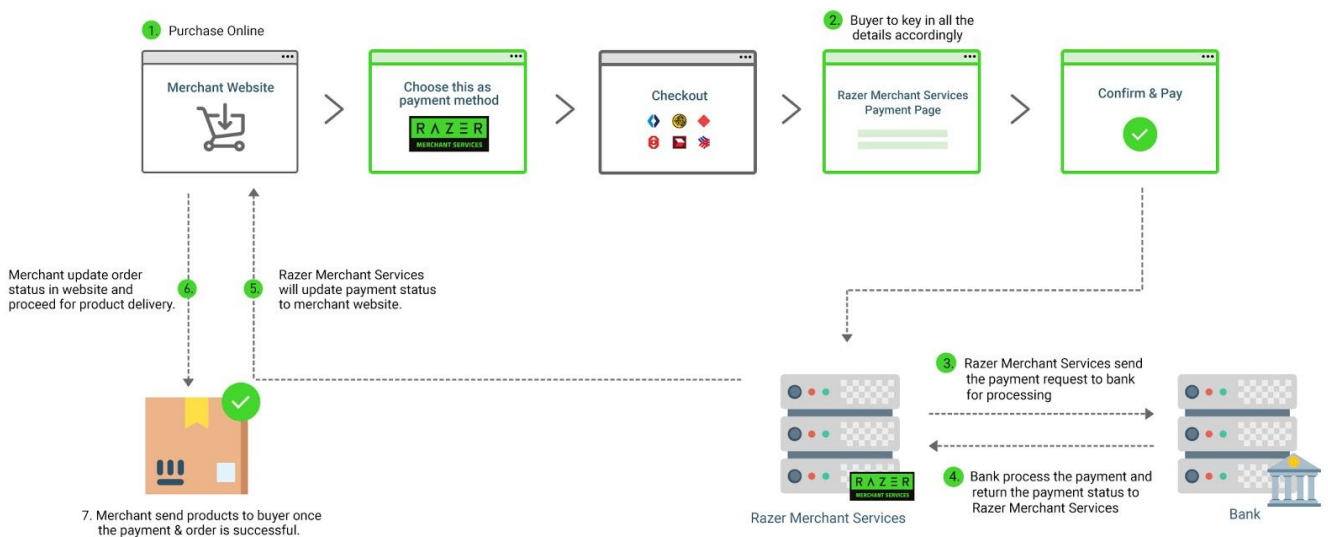
**Razer Merchant Services (RMS)** is to help merchant, who sell online and offline, to reach out to the regional buyers, collecting payment and distributing funds, expanding rapidly to South-East Asia market. The service includes:

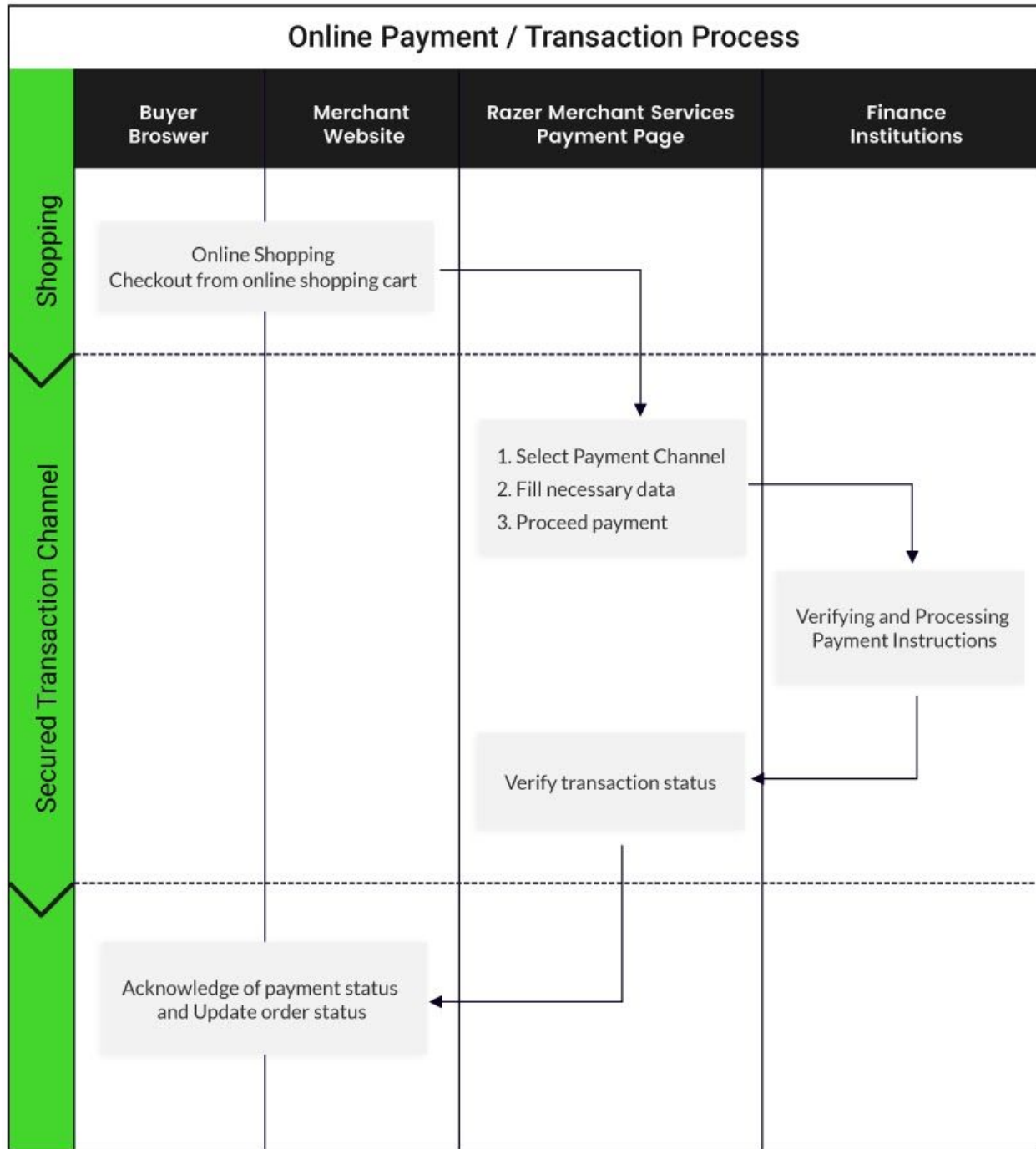
- Front-end
  - RWD or responsive web design payment page for online buyer to checkout
  - Secure 1-click payment using credit card tokenization technology, to ease the payment flow for returning purchase
  - Bcard redemption and rewarding loyalty program for buyers
  - Seamless integration to popup bank login page immediately
  - In-page checkout similar to Stripe and airbnb checkout flow that can seamlessly integrated with merchant system
  - Mobile XDK allows app developers to implement in-app payments
  - Channel switching is available for same currency channels
  - Razer Cash which allows over-the-counter (OTC) payment acceptance
  - Common shopping carts payment module, plugin, addon, or extension supported
- Back-end
  - Server-to-server Notification to ensure no missing status update
  - Callback to update merchant system on deferred status change
  - Merchant can login to control panel to track payment status
  - Scheduled report on daily/weekly/monthly basis to update merchant via email
  - Real-time visualized reports

# PAYMENT FLOW OVERVIEW

RMS provides hosted payment page service, so that merchant does not require costly and tech-savvy PCI-DSS compliance at merchant website or system. The integration is as simple as passing parameters via **HTTPS POST** or **GET** method from merchant to RMS payment page. Buyer will proceed their transaction on internet banking or any payment channel. Once completed, RMS will redirect buyer's front-end back to merchant system, using **POST** method.

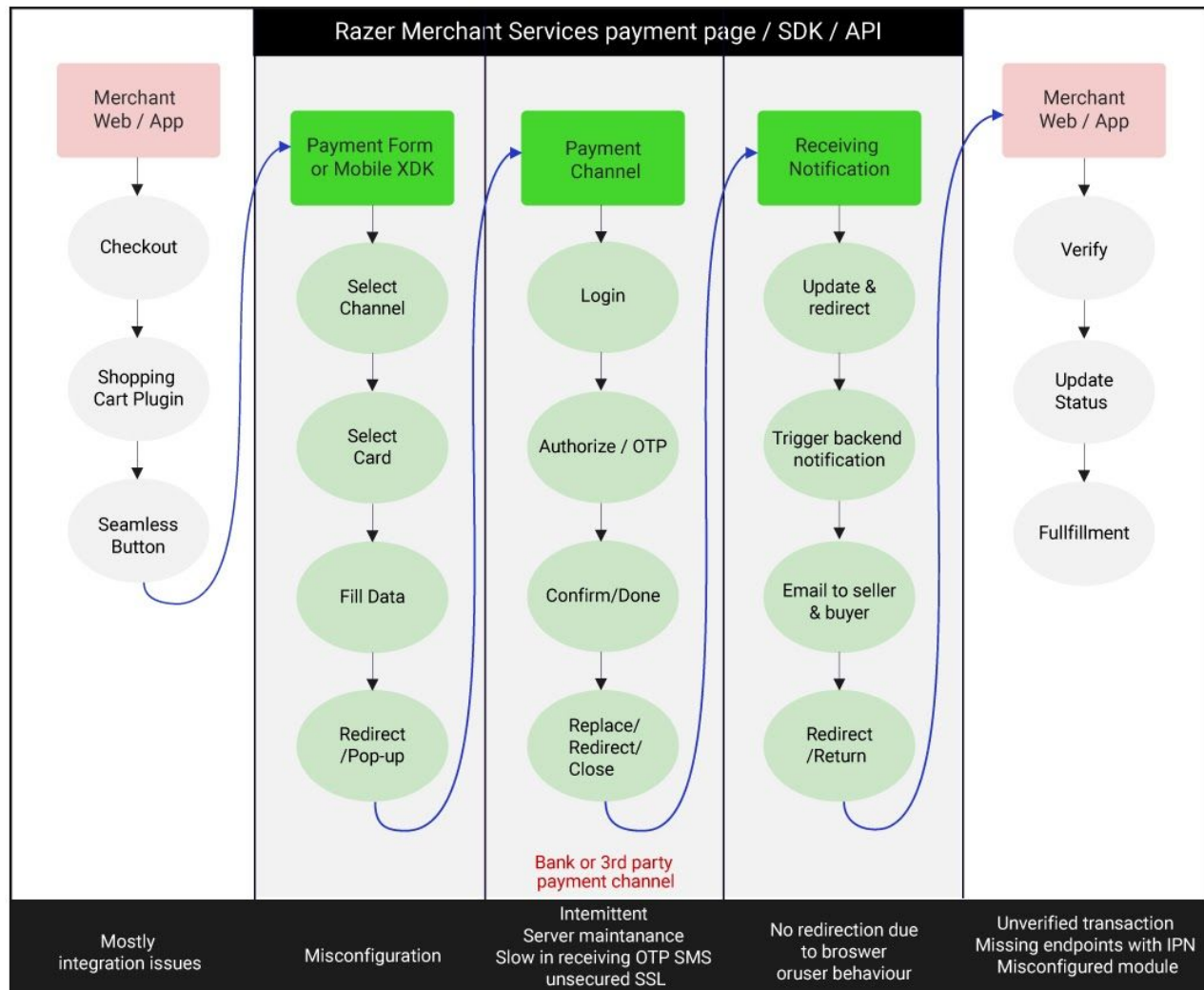
**IPN** (instant payment notification) or **ACK** from merchant could be implemented to confirm the receiving of payment status update.







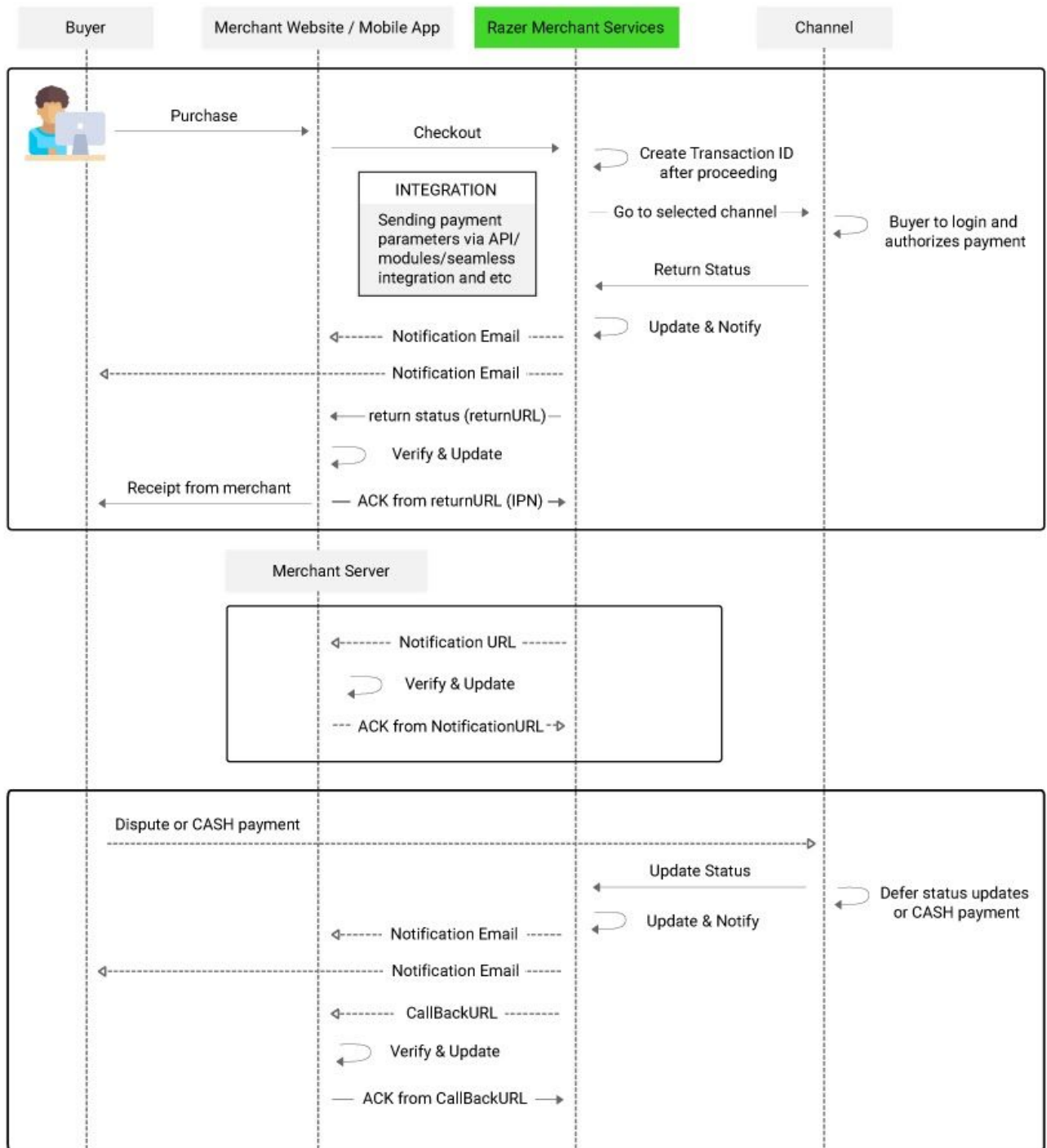
## How to identify and report tech-related issue to RMS?



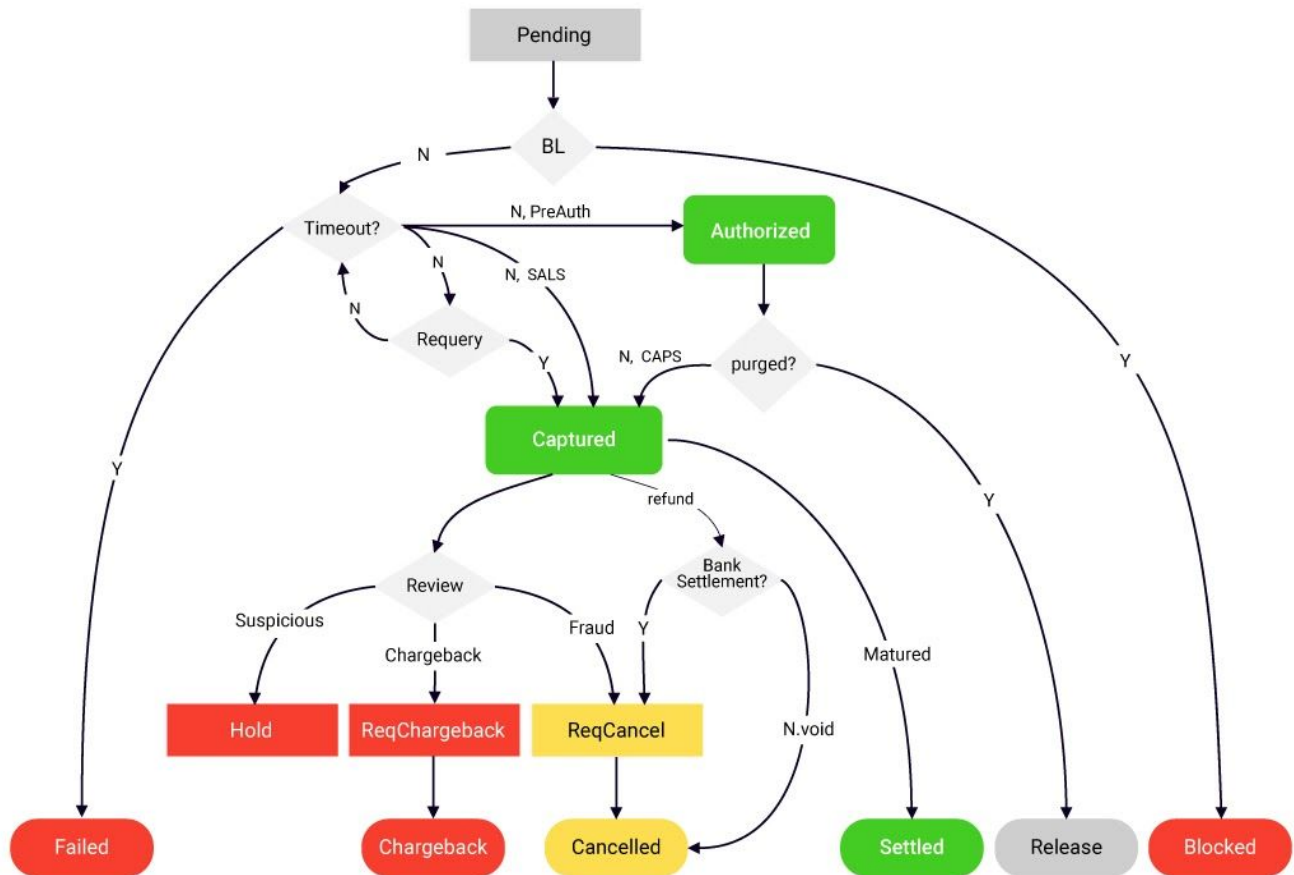
Merchant needs to prepare 3 simple and similar payment endpoint scripts to handle the payment notification from the payment gateway:

1. **return URL:** front-end or browser redirection notification, which are normally not a 100% reliable and robust channel due to unexpected network connectivity issue or client-side behavior, such as browser application crashes;
2. **notification URL:** a server-to-server back-end webhook which is more robust and crucial, especially for mobile application;
3. **callback URL:** a special handler webhook to get notified on non-realtime payment status, such as “deferred status update”, change of payment status, or Razer Cash, which is not a realtime payment naturally.

After the normal payment flow, merchant can always send payment status query request, which is defined in **reQuery** APIs (a.k.a PSQ, Payment Status Query).



## Razer Merchant Services Payment Status Flow



# SECURITY & DATA INTEGRITY

For online payment. The system uses “**merchant ID**”, “**Verify Key**”(public key, a.k.a verification key) and “**Secret Key**”(private key) to generate encrypted hash string to ensure data integrity in the payment process.

## Verify Key (Public Key)

Verify Key is unique shared secret for RMS merchants to generate request to the payment gateway. It is a key or seed for generating one-time hash data, which are known as “**vcode**” (merchant’s payment request to the gateway) or some of the “**skey**” (APIs’ request from merchant to the gateway).

### How to get the verify key?

1. Logon to RMS Merchant Portal website
2. Go to Merchant Profile tab.
3. Scroll down until you see the word “Verify Key”
4. Get the value and use it on any functions that require it.

Multi-Payment Channel	<input type="checkbox"/> Enable <span>(iv)</span>
Verify Key	XXXXXXXXXXXXXXXX
Enable Verify Payment	<input type="checkbox"/> Tick to enable transaction amount and order number verification <span>(v)</span>

XXXXXXXXXXXXXXXX is Merchant’s Verify Key provided by the payment gateway. Please make sure it is at least 32 characters. Merchant may request to change the key when necessary.

## Secret Key (Private Key)

Secret Key is unique shared secret for the payment gateway to generate response to merchant. It is a key or seed for generating one-time hash data, which are known as “**skey**” (in the payment response) or some of the “**VrfKey**” (APIs’ response from payment gateway). Merchant or developer is advised **NOT** to disclose this secret key to the public. Once the key is suspected to be compromised, please contact RMS customer service immediately to reset the key.

Merchants who are using RMS mobile xdk MUST use different strings for “Verify Key”(public key) and “Secret Key”(private key), respectively. Contact RMS customer service to reset your keys whenever needed.

## vcode (in payment request)

**vcode** is to ensure the data integrity passed from merchant-end (either website or mobile APP) to the payment page to avoid man-in-the-middle (MITM) attack. It uses "Verify Key"(public key) in combination of the data string for hashing purpose.

It becomes mandatory for each transaction if "Enable Verify Payment" is activated in merchant profile as shown:-

Enable Verify Payment	<input checked="" type="checkbox"/>	Tick to enable transaction amount and order number verification (v)
Notified Failure Transaction	<input type="checkbox"/>	Tick to receive a copy of failed transaction notification.
Targeted Market	<input type="checkbox"/>	Allow transaction made using Malaysia IP Address only

**vcode** was encrypted using MD5 encryption hash function and consists of the following information (must be set in the following orders) :

1. Transaction amount
2. Merchant ID
3. Order ID
4. Verify Key

### Formula to generate **vcode**

$$vcode = md5( \text{amount} \ \& \ \text{merchantID} \ \& \ \text{orderId} \ \& \ \text{verify\_key} )$$

### Example to generate vcode for PHP developer

```
<?php
$amount      = "27.60";
$merchantID = "ACME";
$orderid     = "OD8842";
$verifykey   = "xxxxxxxxxxxxxxxxxxxx";
// Replace xxxxxxxxxxxxxxxxxxxx with your Verify Key

// vcode formula
$vcode       = md5( $amount.$merchantID.$orderid.$verifykey );

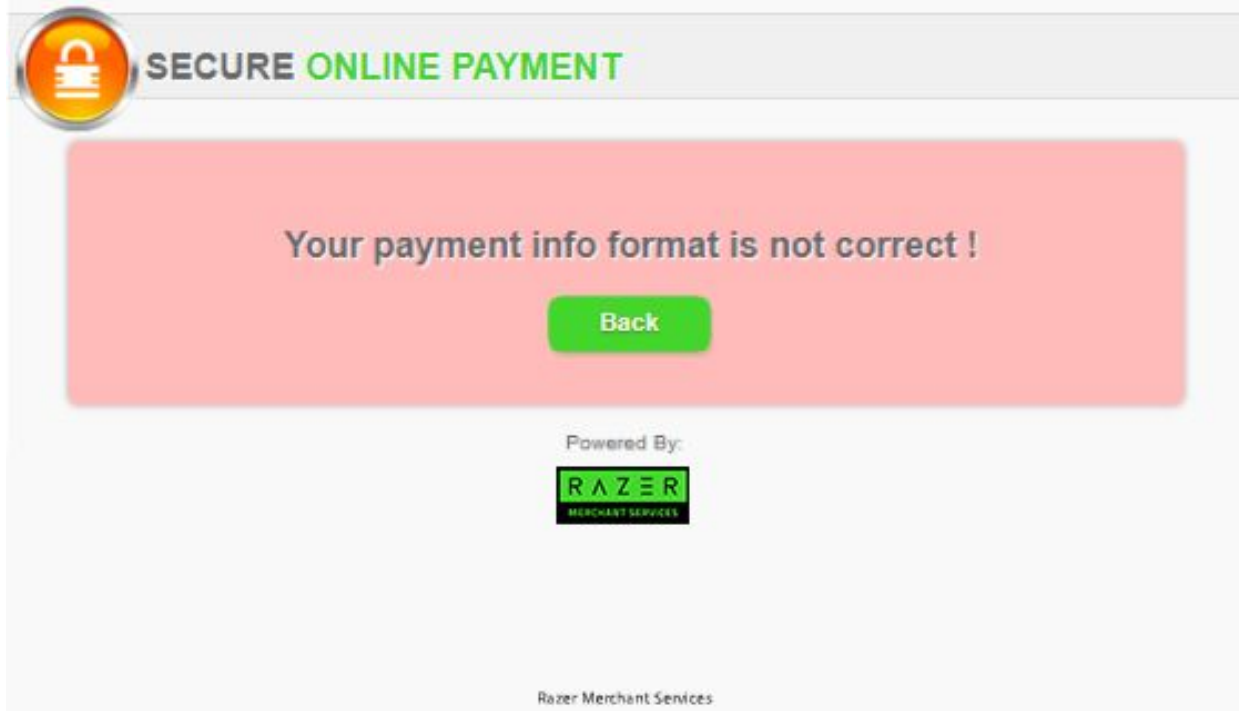
// output of the vcode based on above information equals to :
$vcode       = "ec7f2c6e85769728a5e9b75893ee6bc1";
?>
```

### Verification tool for vcode

To verify whether the **vcode** generated is correct, merchant may check on this URL:-  
<https://api.molpay.com/MOLPay/query/vcode.php>

### What happen if merchant passes in an incorrect **vcode**?

An error will be displayed on the payment page as shown:-



## skkey (in payment response)

**skkey** is a payment gateway generated returning hash string to ensure the payment result data integrity that passed to merchant system. Merchant MUST verify this hash string properly and compare the order ID, currency, amount, and also the payment date/time, to protect self interest from being cheated by fraudster/hacker. It uses "Secret Key"(private key) in combination of data string for hashing process.

Please note that there are other "**skkey**"s with different formula in **Merchant Request APIs'** parameter. Try not to confuse yourself with this payment response "**skkey**".

**skkey** was encrypted twice using MD5 encryption hash function and consists of the following information (must be set in the following orders) :

### First hash string

1. Transaction ID
2. Order ID
3. Status
4. Merchant ID (domain)
5. Amount
6. Currency

### Final hash string

1. Payment Date/Time
2. Merchant ID (domain)
3. First hash string
4. Approval Code
5. Secret Key

### Formula to generate skkey

$pre\_skkey = md5( txnID \& orderID \& status \& merchantID \& amount \& currency )$

$skkey = md5( paydate \& merchantID \& pre\_skkey \& appcode \& secret\_key )$

### Example to generate skkey for PHP developer

```
<?php

$skkey = "xxxxxxxxxx"; //Replace xxxxxxxxxxxx with your Secret_Key

/*****
*Don't change below parameters
*****/
$tranID = $_POST['tranID'];
$orderid = $_POST['orderid'];
$status = $_POST['status'];
$merchant = $_POST['domain'];
$amount = $_POST['amount'];
```

```

$currency      =      $_POST['currency'];
$appcode       =      $_POST['appcode'];
$paydate       =      $_POST['paydate'];
$key           =      $_POST['key']; //Security hashstring returned by PG

/*****
* To verify the data integrity sending by PG
*****/
$key0 = md5( $tranID.$orderid.$status.$merchant.$amount.$currency );
$key1 = md5( $paydate.$merchant.$key0.$appcode.$vkey );
//key1 : Hashstring generated on Merchant system
// either $merchant or $domain could be one from POST
// and one that predefined internally
// by right both values should be identical

if( $key === $key1 ){
    // If matched, perform another extra checking before approved order
} elseif( $key != $key1 ){
    // If unmatched, reject the order or merchant might send query to
    // PG using Merchant requery to double check payment status
    // for that particular order.
} else {
    // error or exception case
}

```

?>



# SIMULATION

Merchant should be provided a sandbox account or developer account from RMS, in order to simulate the payment flow for successful and failure transaction cases.

Please note that sandbox account is for **SIT** purpose and only faked credit card channel is available. **Sandbox** environment is solely for basic integration and validation purpose, **DO NOT** treat it as a grant for production environment testing and the configuration and implementation might be different. “Dev” account has full range of production feature for advanced integration and **UAT** purpose.

## Sandbox Account

- Sandbox provides only simple simulation on card payment for merchant to test successful and failure cases based on algorithm (Luhn-10) and no actual transaction occurs between bank or payment gateway
- Many advanced features are not available in the sandbox environment
- Test card - sample for Visa and Mastercard card number for testing:

Pattern	MasterCard	Visa
Positive Test	5105105105105100 (CVV: 444) 5555555555554444 (CVV: 444)	4111111111111111 (CVV: 111) 4012888888881881 (CVV: 111)
Negative Test	5555555555554440 (CVV: 444)	4111111111111110 (CVV: 111)

- Expiry date for credit card need to be valid (NOT expired) as of the current month and year.

## Developer Account

For full feature integration and **UAT**, such as mobile xdk or seamless integration, please request “Developer” account or “Dev” account from RMS. These accounts are all set with a short period of lifespan for integration and testing purpose only. No settlement and mass payout is allowed in these accounts.

For real transaction conducted in “Dev” account, please void or cancel the transaction on the same day. For Malaysian internet banking, only Maybank, CIMB Clicks, and FPX offers refund feature. You may request to turn on the channel if it is not enabled in your “Dev” account.

A “Dev” or developer account is always ended in “\_Dev” and there will be **no settlement** (hence no payout) for this type of account. Please make sure that your “Dev” account have the same configuration in the production merchant profile to guarantee the expected test results.

# PAYMENT APIs

## Initiating Payment

Passing parameters to the payment gateway hosted payment page using POST/GET method via HTTPS or SSL connection will initiate a payment request from merchant system.

### REQUEST URL

Production: [https://www.onlinepayment.com.my/MOLPay/pay/MerchantID/Payment\\_Method](https://www.onlinepayment.com.my/MOLPay/pay/MerchantID/Payment_Method)

Sandbox: [https://sandbox.molpay.com/MOLPay/pay/MerchantID/Payment\\_Method](https://sandbox.molpay.com/MOLPay/pay/MerchantID/Payment_Method)

**MerchantID**: mandatory value to identify merchant;

**Payment\_Method**: optional value to identify payment channel; Payment channel can be identified by using **channel filename**, or pass the **channel parameter** to **index.php** using GET/POST method.

The URL is an API to accept POST/GET parameters from merchant site as well as the payment page for buyer. For merchant who uses POST method please include the " / " after **MerchantID** on the URL.

WARNING: Loading the bank login page or authorization page into iframe is not recommended as most of the banks or channels will block all cross-origin requests and treat it as XFS, XSS, and CSRF attack.

## Channel Lists

### Multi-Currency Payment

Default currency: -

Channel Name	filename	channel	Payable Amount Range	Extra Information
Visa / MasterCard	indexT.php	creditT	> 1.00	51 currencies supported : AED, <b>AUD</b> , <b>BND</b> , BRL, CAD, CHF, <b>CNY</b> , COP, CZK, DKK, EGP, <b>EUR</b> , FJD, <b>GBP</b> , <b>HKD</b> , HUF, <b>IDR</b> , ILS, INR, IQD, IRR, ISK, <b>JPY</b> , <b>KRW</b> , KWD, LAK, LKR, MOP, MXN, <b>MYR</b> , NGN, NOK, NZD, <b>PHP</b> , PKR, PLN, QAR, RUB, SAR, SCR, SEK, <b>SGD</b> , <b>THB</b> , TRY, <b>TWD</b> , <b>USD</b> , <b>VND</b> , ZAR, BDT, NPR, MMK

### Malaysia

Default currency: MYR

Channel Name	channel filename	channel parameter	Payable Amount Range / Downtime*	Extra Information
Visa / MasterCard	index.php	credit	> 1.00	credit/debit/prepaid card
<b>FPX B2C</b>				
FPX B2C Bank Islam	BIMB.php	BIMB	> 1.00	indirect internet banking
FPX B2C Maybank	MB2U.php	MB2U	> 1.00 / 12am - 12:30am	indirect internet banking
FPX B2C Public Bank	PBB.php	PBB	> 1.00	indirect internet banking
FPX B2C CIMBClicks	CIMBCLICKS.php	CIMBCLICKS	> 1.00	indirect internet banking
FPX B2C AmOnline	AMOnline.php	AMOnline	> 1.00	indirect internet banking
FPX B2C Hong Leong	HLBConnect.php	HLBConnect	> 1.00	indirect internet banking
FPX B2C RHB Bank	RHBNow.php	RHBNow	> 1.00 / 12am - 12:10am	indirect internet banking
FPX B2C OCBC	FPX_OCBC.php	FPX_OCBC	> 1.00	indirect internet banking
FPX B2C Standard Chartered	FPX_SCB.php	FPX_SCB	> 1.00	indirect internet banking
FPX B2C Affin Bank	FPX_ABB.php	FPX_ABB	> 1.00	indirect internet banking
FPX B2C Bank Rakyat	bankrakyat.php	bankrakyat	> 1.00 / 12am - 12:30am	indirect internet banking
FPX B2C Bank Muamalat	bankmuamalat.php	bankmuamalat	> 1.00	indirect internet banking

FPX B2C Kuwait Finance House	FPX_KFH.php	FPX_KFH	> 1.00	indirect internet banking
FPX B2C Bank Simpanan Nasional	FPX_BSN.php	FPX_BSN	> 1.00 / 12am - 12:15am	indirect internet banking
FPX B2C Alliance Bank	FPX_ABMB.php	FPX_ABMB	> 1.00	indirect internet banking
FPX B2C United Overseas Bank	FPX_UOB.php	FPX_UOB	> 1.00	indirect internet banking
<b>FPX B2B</b>				
FPX B2B AmBank	FPX_B2B_AMB.php	FPX_B2B_AMB	2.00 - 1mil / 12am - 2am	indirect internet banking
FPX B2B Hong Leong	FPX_B2B_HLB.php	FPX_B2B_HLB	2.00 - 1mil	indirect internet banking
FPX B2B United Overseas Bank	FPX_B2B_UOB.php	FPX_B2B_UOB	2.00 - 1mil	indirect internet banking
FPX B2B Affin Bank	FPX_B2B_ABB.php .php	FPX_B2B_ABB	2.00 - 1mil	indirect internet banking
FPX B2B CIMB Bank	FPX_B2B_CIMB.php	FPX_B2B_CIMB	2.00 - 1mil	indirect internet banking
FPX B2B Bank Islam	FPX_B2B_BIMB.php	FPX_B2B_BIMB	2.00 - 1mil / 11pm - 7am	indirect internet banking
FPX B2B Public Bank	FPX_B2B_PBB.php	FPX_B2B_PBB	2.00 - 1mil	indirect internet banking
FPX B2B RHB Bank	FPX_B2B_RHB.php	FPX_B2B_RHB	2.00 - 1mil / 12am - 12:15am	indirect internet banking
FPX B2B HSBC	FPX_B2B_HSBC.php	FPX_B2B_HSBC	2.00 - 1mil	indirect internet banking
FPX B2B Maybank2E	FPX_M2E.php	FPX_M2E	2.00 - 1mil / 12am - 1 am	indirect internet banking
FPX B2C DirectDebit e-Mandate	FPX_EMANDATE.php	FPX_EMANDATE	> 1.00 Maximum authorized installment amount but always debit MYR 1.00 upon enrollment and this will be fully refunded	direct debit enrollment
<b>DIRECT IB</b>				
Maybank2u	maybank2u.php	maybank2u	> 1.00 / 12:15am - 1am	direct internet banking
CIMB Clicks	cimb.php	cimb	> 1.00 / 11:45pm - 12:45am	direct internet banking / CIMB card
Bank Islam	BIMB.php	BIMB	> 1.00 / 12:15am - 1am	direct internet banking
RHB Now	rhb.php	rhb	> 1.00 / 12am - 1am	direct internet banking / RHB card

Am Online	amb.php	amb	> 1.00 / 12am - 6am	direct internet banking
Hong Leong Connect	hlb.php	hlb	> 1.00 / 11pm - 7am	direct internet banking / HLB card
PEX+ by HLBB	PEXPLUS.php	PEXPLUS	> 1.00	direct internet banking
Affin Bank	affin-epg.php	affin-epg	> 2.00	direct internet banking
<b>Wallet</b>				
<b>Razer Pay</b>	RazerPay.php	RazerPay	> 1.00	e-wallet
<b>Razer Gold</b>	MOLPoints.php	MOLPoints	> 1.00	e-wallet
Webcash	webcash.php	webcash	> 1.00	e-wallet
Touch N' Go E-WALLET	TNG-EWALLET.php	TNG-EWALLET	> 1.00	e-wallet
BOOST	BOOST.php	BOOST	> 1.00	e-wallet
GrabPay	GrabPay.php	GrabPay	> 1.00	e-wallet
Maybank QRPAY	MB2U_QRPAY-Push.php	MB2U_QRPAY-Push	> 1.00	e-wallet
<b>OTC</b>				
<b>Razer Cash</b> via 7E	cash.php	cash	> 1.00	OTC @ 7-Eleven
JomPAY ATM 9282	jompay.php	jompay	> 1.00	ATM bill payment
CIMB Virtual Account	CIMB-VA.php	CIMB-VA	> 1.00	CIMB Clicks / ATM transfer for CIMB customers only
<b>Razer Cash</b> via 99	cash99.php	cash99	> 1.00	OTC @ 99 Speedmart
<b>Installment for card</b>				
MBB-eBPG-(PD or PW) EzyPay	MBB-EzyPay.php	MBB-EzyPay	Non-installment > 1.00 Installment: 3 mon: 200 - 1000 6 mon: 500 - 15,000 12 mon: 500 - 15,000 24 mon: 2,500 - 15,000	MBB card installment V/M card payment
PBB-ZIIP	installmentPBB-ZIIP.php	PBB-ZIIP	non-installment > 1.00 installment > 500.00	PBB card installment V/M card payment
CIMB-eBPG	indexL.php	creditL	non-installment > 1.00 installment > 500.00	CIMB card installment V/M card payment
AMBANK-eBP-PD	indexU.php	creditU	non-installment > 1.00 installment > 500.00	AMB card installment V/M card payment

\* Downtime is for reference only

#### Remarks

1. Secure 1-click or credit card tokenization function can be enabled in merchant profile to ease next purchase on the same merchant system.
2. Bcard loyalty reward points for successful transaction can be activated in merchant profile. (Obsoleted)

- For certain Razer Cash channels, such as 7-eleven and epay, round to the nearest 0 or 5 cents and convenience store fee applied. Buyer will see the "Payable Amount", but RMS still return the amount passed from merchant system.

## Singapore

**Default currency: SGD**

Channel Name	filename	channel	Payable Amount Range	Extra Information
eNETS Debit	enetsD.php	enetsD	> 5.00	indirect internet banking
SAM by SingPost	singpost.php	singpost	> 5.00	bank card / ATM
AXS	AXS.php	AXS	> 5.00	bank card / ATM
Wallet				
Razer Pay	RazerPay.php	RazerPay	> 1.00	e-wallet

## Indonesia

**Default currency: IDR**

Channel Name	filename	channel	Payable Amount Range	Extra Information
Artajasa	Artajasa.php	Artajasa	> 10,000.00	OTC
ATM Network Transfer Virtual Account (VA)	ATMVA.php	ATMVA	> 10,000.00 <= 10,000,000.00	cardless ATM

## China

**Default currency: USD/MYR/SGD (for merchant), RMB/CNY/MYR/SGD/USD (for buyer)**

Channel Name	filename	channel	Payable Amount Range (Processing Currency)	Extra Information
Alipay 支付宝	alipay.php	alipay	> MYR1.00 (MYR/USD/SGD/RMB/CNY)	e-wallet (CN)
UnionPay 银联	GUPOP.php	GUPOP	> MYR1.00 (MYR/USD/CNY/AUD/CAD/EUR/GBP/HKD/IDR/JPY/NZD/PHP/SGD/THB/TWD/VND)	bank card
WeChat Pay 微信支付	WeChatPay.php	WeChatPay	> USD1.00	e-wallet (CN)

### Remarks

- Due to China Central Bank regulations, CNY(or RMB) is not able to settle to oversea bank account, these channels will settle USD as default settlement currency. Beside USD, RMS can also settle MYR, SGD via Alipay as well and RMS will convert it to equivalent amount of settlement currency based on Maybank daily FX rate.

## Thailand

Default currency: THB

Channel Name	filename	channel	Payable Amount Range	Extra Information
Bank of Ayudhya (Krungsri)	BAY_IB_U.php	BAY_IB_U	> 1.00	direct internet banking (fees on buyer)
Bangkok Bank	BBL_IB_U.php	BLL_IB_U	> 1.00	direct internet banking (fees on buyer)
Krung Thai Bank	KTB_IB_U.php	KTB_IB_U	> 1.00	direct internet banking (fees on buyer)
Tesco Lotus via Omise	OMISE_TL.php	OMISE_TL	> 20.00	OTC
Big Central	BigC.php	BigC	> 1.00	OTC
Kasikornbank PAYPLUS	KBANK_PayPlus.php	KBANK_PayPlus	> 1.00	direct internet banking (fees on buyer)
Siam Commercial Bank	SCB_IB_U.php	SCB_IB_U	> 1.00	direct internet banking (fees on buyer)

## Vietnam

Default currency: VND

Channel Name	filename	channel	Payable Amount Range	Extra Information
NganLuong	nganluong.php	nganluong	> 2,000.00	e-wallet
VTC Pay	vtcpay.php	vtcpay	> 1,000.00	e-wallet

## Philippines

Default currency: PHP

Channel Name	filename	channel	Payable Amount Range	Extra Information
Dragonpay	dragonpay.php	dragonpay	> 50.00	indirect internet banking / cash

## Taiwan

Default currency: TWD

Channel Name	filename	channel	Payable Amount Range	Extra Information
ESUN Cash-711	ESUN_Cash711.php	ESUN_Cash711	>= 1.00	OTC
ESUN Cash FamilyMart	ESUN_CashFamilyMart.php	ESUN_CashFamilyMart	>= 1.00	OTC
ESUN ATM	ESUN_ATM.php	ESUN_ATM	>= 1.00	Virtual Account
ESUN Cash Hi-Life	ESUN_CashHiLife.php	ESUN_CashHiLife	>= 1.00	OTC



## Payment Page Integration

This is the traditional integration method which will send the buyer information to the payment gateway hosted payment page. You may want to explore *seamless integration*, *Direct Server API*, *mobile xdk* or supported shopping cart payment modules for better user experience beside this hosted payment page integration on github repositories. Recurring API is also available for card payment and DirectDebit.

### Request Parameters

These parameters can be passed using either POST or GET method, or mixed of these methods. Please use UTF-8 encoding for all values.

Variable / Parameter	Type Format / Max Length	Description / Example
merchant_id	optional, alphanumeric	Merchant ID provided and it is on the request URL, otherwise need to pass this as parameter
amount	mandatory, integer or up to 2 decimal points numeric value	The total amount to be paid in one purchase order. Configurable to lock this field (Read-only). E.g. 500, 168.99, comma(,) is not allowed
orderid	mandatory, alphanumeric up to 32 characters	Invoice or order number from merchant system. Can set to Read-only field. E.g. BH2018-09rev
bill_name	mandatory, alphanumeric, 128 chars	Buyer's full name.
bill_email	mandatory, email, 128 chars	Buyer's email address. Must be valid email.
bill_mobile	mandatory, alphanumeric, 32 chars	Buyer's mobile number or contact number. Must be valid mobile number.
bill_desc	mandatory, alphanumeric, 64kB	Purchase itemized list or order description. Try to avoid special character so that the payment request is not blocked by web application firewall. For Taiwan channels please refer to <b>Guideline for Taiwan channels</b> .
country	mandatory, 2 chars of ISO-3166 country code (Alpha-2)	Buyer's country E.g. MY for Malaysia.
vcode	conditional if accept open amount or order payment, such as virtual terminal is optional otherwise is mandatory. 32 chars hexadecimal string	This is the data integrity protection hash string. Refer <b>vcode</b> section for details.
cur / currency	optional, 3 chars ISO-4217 currency code	Default payment currency from merchant site. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
channel	optional, predefined string. Refer to <i>channel</i> column in the Channel Lists.	Default payment page will be displayed without <i>channel</i> specified.
langcode	optional, predefined string. Currently only <i>en</i> for English & <i>cn</i> for Simplified Chinese.	Default language, i.e. English, will be displayed without langcode specified.

returnurl	optional, URL	For selected merchant only. Used for multiple return URL. All URLs must be registered beforehand with RMS. This will slow down your payment request.
callbackurl	optional, URL	Used for both multiple callback URL and notification URL. All URL endpoints must be registered beforehand with RMS.
cancelurl	optional, URL	Buyer will redirect to this URL if buyer click "Cancel" button to abandon payment before proceeding. No transaction will be created.
state	optional, mandatory for specific channel	Buyer's state
zipcode	optional, mandatory for specific channel	Buyer's zipcode or postcode
addr	optional, mandatory for specific channel	Buyer's address
s_country	optional, mandatory for specific channel	Delivery destination country
s_state	optional, mandatory for specific channel	Delivery destination state
s_zipcode	optional, mandatory for specific channel	Delivery destination zipcode
s_addr	optional, mandatory for specific channel	Delivery destination address
s_merchantID	conditional for partner	Partner's sub-merchant MID
s_merchantURL	conditional for partner	Partner's sub-merchant URL
s_merchantMCC	conditional for partner	Partner's sub-merchant MCC
is_escrow	optional, for escrow payment	Set the value to 1 to indicate this is an escrow payment. Escrow transaction also can be set after the transaction has been created using escrow API.
non_3DS	conditional for partner, 0 or 1	Applicable to card processing via specific processor using specific currency for pre-approved partner only. Equal to 0 by default and 1 for non-3DS transaction
tcctype	optional, for card type transaction	Available value is: SALS = Capture Transaction (Default) AUTH = Authorize Transaction (Please inform RMS before starting using pre-auth)
token	optional	Token value for 1-click payment to show the selected card
installmonth	optional, mandatory for installment payment	Total month of installment. E.g: 0, 3, 6, 12, 24
cash_waittime	optional, integer	To overwrite the allowed waiting time for cash payment in hour(s), value greater than "Payment Expiry Time" in merchant profile will be capped.
split_info	optional, alphanumeric with pipe (   ) as data delimiter and comma ( , ) as recipient separator	Used for "Alipay Split Payment". Format as below without '<' and '>':- <submer_ID> <amount>, <submer_ID> <amount>

## Examples

### Using PHP to generate a GET payment request

```
<?PHP
echo "<a href=https://www.onlinepayment.com.my/MOLPay/pay/merchantID/?";
echo "amount=".$amount."&";
echo "orderid=".urlencode($oid)."&";
echo "bill_name=".urlencode($name)."&";
echo "bill_email=".urlencode($email)."&";
echo "bill_mobile=".urlencode($mobile)."&";
echo "bill_desc=".urlencode($description)."&";
echo "country=".$country."&";
echo "vcode=".$vcode."> Pay Now </a>";
?>
```

### Using PHP to generate a POST form payment request

```
<?PHP
echo "<form action=https://www.onlinepayment.com.my/MOLPay/pay/merchantID/' method=POST >";
echo "<input type=hidden      name=amount          value='$amount'>";
echo "<input type=hidden      name=orderid          value='$oid'>";
echo "<input type=hidden      name=bill_name         value='$name'>";
echo "<input type=hidden      name=bill_email        value='$email'>";
echo "<input type=hidden      name=bill_mobile       value='$mobile'>";
echo "<input type=hidden      name=bill_desc         value='$description'>";
echo "<input type=hidden      name=country           value='$country'>";
echo "<input type=hidden      name=vcode             value='$vcode'>";
echo "<input type=submit      value=' PAY NOW '>";
echo "</form>";
?>
```

### Using ASP to generate a GET payment request

```
<%
dim amount,orderid,name,email,mobile,desc,country,vcode
amount  = "18.99"
orderid= "DG873MH370"
name    = "Mr Albert Anderson"
email   = "a.anderson@somewhere.com"
mobile  = "016-2341234"
desc    = "DIGI Reload Coupon RM20 with discount"
country= "MY"
vcode   = md5(amount & merchantID & orderid & xxxxxxxxxxxx )
// REPLACE xxxxxxxxxxxx with Verify Key
response.write("<a href=https://www.onlinepayment.com.my/MOLPay/pay/merchantID/?")
response.write("&amount=" & amount)
response.write("&orderid=" & orderid)
response.write("&bill_name=" & name)
```

```

response.write("&bill_email=" & email)
response.write("&bill_mobile=" & mobile)
response.write("&bill_desc=" & desc)
response.write("&country=" & country)
response.write("&vcode=" & vcode)
response.write("> Pay Now </a>")

```

%>

## Using ASP to generate a POST form payment request

```

<%
dim amount, orderid, name, email, mobile, desc, country, vcode
amount = "28.99"
orderid= "DG873MH37o"
name = "Mr Samuel Lim"
email = "sam.lim@nowhere.com"
mobile = "86-232389872"
desc = "Facebook Prepaid Reload Coupon RM30 with discount"
country= "CN"
vcode = md5(amount & merchantID & orderid & xxxxxxxxxxxx )
// REPLACE xxxxxxxxxxxx with Verify Key
response.write("<form action='https://www.onlinepayment.com.my/MOLPay/pay/merchantID/' method=POST >")
response.write("<input type=hidden name=amount value='&amount'>")
response.write("<input type=hidden name=orderid value='&orderid'>")
response.write("<input type=hidden name=bill_name value='&name'>")
response.write("<input type=hidden name=bill_email value='&email'>")
response.write("<input type=hidden name=bill_mobile value='&mobile'>")
response.write("<input type=hidden name=bill_desc value='&desc'>")
response.write("<input type=hidden name=country value='&country'>")
response.write("<input type=hidden name=vcode value='&vcode'>")
response.write("<input type=submit value=' PAY NOW ' ">")
response.write("</form>")

```

%>

## Getting Payment Result

Payment result will be returned to merchant system once payment is done or user abandons the payment process. HTTP POST is the only method that payment gateway returns all parameters to merchant's return URL for real-time status update, which merchant can configure it in merchant admin. Merchant system should block all other methods or parameters from untrusted source.

### Response Parameters

Variable / Parameter	Type Format / Max Length	Description / Example
amount	2 decimal points numeric value	The total amount paid or to be paid in Razer Cash payment request.
orderid	alphanumeric, 32 characters	Invoice or order number from merchant system.
tranID	integer, 10 digits	Unique transaction ID for tracking purpose.
domain	alphanumeric, 32 chars	Merchant ID in RMS PG system.
status	2-digit numeric value	00 for Successful payment, 11 for failure, 22 if pending.
appcode	alphanumeric, 16 chars	Bank approval code. Mandatory for card payment. Certain channel returns empty value.
error_code	alphanumeric	Refer to the Error Codes section.
error_desc	text	Error message or description.
skey	32 chars hexadecimal string	This is the data integrity protection hash string. Refer <a href="#">skey</a> section for details.
currency	2 or 3 chars (ISO-4217) currency code	Default currency is MYR (indicating Malaysia Ringgit) for Malaysia channels.
channel	predefined string in PG system	Channel references for merchant system. Refer to the below table.
paydate	Date/Time( YYYY-MM-DD HH:mm:ss)	Date/Time of the transaction.
extraP*	optional (on request)	JSON encoded string or array  token: 16-digit card token for merchant to store for recurring MIT (merchant initiated transaction)  fraudscreen: 1-digit integer, i.e. 1=Unknown, 2=Passed, 3=Alert, 4=Suspicious, 5=Fraud  ccbrand: Visa, MasterCard, AMEX bank_issuer: The issuing bank or institution ccbin: Issuer/Bank identification number, first 6-digit cclast4: Last 4-digit of card number cctype: Credit, Debit, Prepaid

Value of “channel” in return URL

Card / Wallet		
<u>Malaysia &amp; International</u>		
<b>Credit/Debit/Prepaid card (Visa/MasterCard)</b> Credit  <b>Razer zGold - Pay with Razer (PWR)</b> MOLPoints  <b>WeChat Pay MY (Online)</b> WeChatPayMY  <b>Maybank QRPay</b> MB2U_QRPay-Push	<b>Point BCard</b> Point-BCard  <b>Webcash</b> WEBCASH  <b>WeChat Pay MY (Offline)</b> WeChatPayMY-Offline  <b>Touch N' Go E-WALLET</b> TNG-EWALLET	<b>PayPal</b> PayPal  <b>Boost</b> BOOST  <b>GrabPay</b> GrabPay
<u>China</u>		
<b>Alipay (Online)</b> Alipay	<b>WeChat Pay (Online)</b> WeChatPay	<b>WeChat Pay (Offline)</b> WeChatPay-Offline
<u>Thailand</u>		
<b>Paysbuy</b> paysbuy		
<u>Vietnam</u>		
<b>Nganluong</b> NGANLUONG  <b>VTC Pay eWallet &amp; Banks</b> VTCP_EWB	<b>VTC-Pay</b> VTC-Pay  <b>VTC Pay eWallet &amp; Credit Cards</b> VTCP_EWC	<b>VTC Pay eWallet</b> VTCP_EW
Indirect / Direct Internet Banking		
<u>Malaysia</u>		
<b>FPX</b> FPX  <b>RHB Now</b> FPX_RHB / RHB-ONL  <b>Bank Islam</b> FPX_BIMB  <b>Affin Online</b> FPX_ABB / Affin-EPG  <b>myBSN</b> FPX_BSN  <b>HSBC Online</b> FPX_HSBC	<b>Maybank2u</b> FPX_MB2U / MB2u  <b>PBe</b> FPX_PBB / PBeBank  <b>AmOnline</b> FPX_AMB / AMB-W2W  <b>i-Muamalat</b> FPX_BMMB  <b>OCBC Online</b> FPX_OCBC  <b>Standard Chartered Online</b> FPX_SCB	<b>CIMB Clicks</b> FPX_CIMBCLICKS / CIMB-Clicks  <b>Hong Leong Connect / PEx+</b> FPX_HLB / HLB-ONL / PEXPLUS  <b>Alliance online</b> FPX_ABMB / ALB-ONL  <b>i-Rakyat</b> FPX_BKRM  <b>UOB Online</b> FPX_UOB  <b>KFH Online</b> FPX_KFH

<b>FPX B2B Model</b> FPX_B2B  <b>FPX B2B United Overseas Bank</b> FPX_B2B_UOB  <b>FPX B2B Maybank2E</b> FPX_M2E  <b>FPX B2B RHB Bank</b> FPX_B2B_RHB  <b>FPX B2B Deutsche Bank</b> FPX_B2B_DEUTSCHE  <b>FPX B2B OCBC</b> FPX_B2B_OCBC  <b>FPX DirectDebit</b> FPX_DIRECTDEBIT	<b>FPX B2B AmBank</b> FPX_B2B_AMB  <b>FPX B2B Affin Bank</b> FPX_B2B_ABB  <b>FPX B2B CIMB Bank</b> FPX_B2B_CIMB  <b>FPX B2B Public Bank</b> FPX_B2B_PBB  <b>FPX B2B Alliance Bank</b> FPX_B2B_ABMB  <b>FPX B2B Bank Muamalat</b> FPX_B2B_BMMB	<b>FPX B2B Hong Leong</b> FPX_B2B_HLB  <b>FPX B2B HSBC</b> FPX_B2B_HSBC  <b>FPX B2B Bank Islam</b> FPX_B2B_BIMB  <b>FPX B2B Kuwait Finance House</b> FPX_B2B_KFH  <b>FPX B2B Standard Chartered</b> FPX_B2B_SCB  <b>FPX e-Mandate</b> FPX_EMANDATE
<b><u>Singapore</u></b>		
<b>eNets Debit</b> eNets-D		
<b><u>Thailand</u></b>		
<b>Bangkok Bank</b> TH_BBLPN / BBL_IB & BBL_IB_U  <b>Siam Commercial Bank</b> TH_SCBPN	<b>Bank of Ayudhya (Krungsri)</b> TH_BAYPN / BAY_IB_U  <b>OMISE</b> OMISE	<b>Krung Thai Bank</b> TH_KTBPN / KTB_IB & KTB_IB_U
<b><u>Vietnam</u></b>		
<b>VTC Pay Banks</b> VTCP_B  <b>VTC Pay - MB</b> VTCP_MB  <b>VTC Pay - DongABank</b> VTCP_DongABank  <b>VTC Pay - SHB</b> VTCP_SHB  <b>VTC Pay - Eximbank</b> VTCP_Eximbank  <b>VTC Pay - NamABank</b> VTCP_NamABank  <b>VTC Pay - VietABank</b> VTCP_VietABank  <b>VTC Pay - SeaABank</b> VTCP_SeaABank  <b>VTC Pay - GPBank</b> VTCP_GPBank	<b>VTC Pay - Vietcombank</b> VTCP_Vietcombank  <b>VTC Pay - Vietinbank</b> VTCP_Vietinbank  <b>VTC Pay - Oceanbank</b> VTCP_Oceanbank  <b>VTC Pay - VIB</b> VTCP_VIB  <b>VTC Pay - ACB</b> VTCP_ACB  <b>VTC Pay - SaigonBank</b> VTCP_SaigonBank  <b>VTC Pay - VPBank</b> VTCP_VPBank  <b>VTC Pay - PGBank</b> VTCP_PGBank  <b>VTC Pay - BACABANK</b> VTCP_BACABANK	<b>VTC Pay - Techcombank</b> VTCP_Techcombank  <b>VTC Pay - Agribank</b> VTCP_Agribank  <b>VTC Pay - BIDV</b> VTCP_BIDV  <b>VTC Pay - MaritimeBank</b> VTCP_MaritimeBank  <b>VTC Pay - HDBank</b> VTCP_HDBank  <b>VTC Pay - Sacombank</b> VTCP_Sacombank  <b>VTC Pay - TienPhongBank</b> VTCP_TienPhongBank  <b>VTC Pay - Navibank</b> VTCP_Navibank  <b>VTC Pay - PHUONGDONG</b> VTCP_PHUONGDONG

<b>VTC Pay - ABBANK</b> VTCP_ABBANK	<b>VTC Pay - LienVietPostBank</b> VTCP_LVPB	<b>VTC Pay - BVB</b> VTCP_BVB
<u>China</u>		
<b>Union Pay</b> UPOP		
<u>Australia</u>		
<b>POLi Payment</b> PoliPayment		
Kiosk / Over-the-Counter / Cash / Bill Payment		
<u>Malaysia</u>		
<b>7-Eleven</b> Cash-711  <b>Esapay Cash Retail</b> Cash-Esapay  <b>Jompay</b> jompay	<b>99SpeedMart</b> Cash-99SM  <b>Cash-in to PG</b> Cash-Deposit  <b>CIMB Virtual Account</b> CIMB-VA	<b>Petronas</b> Cash-epay  <b>Maybank ATM</b> Cash-MBBATM
<u>Singapore</u>		
<b>SingPost ATM</b> Cash-SAM	<b>AXS Kiosk</b> AXS	
<u>Indonesia</u>		
<b>ATM Transfer via Permata Bank</b> ATMVA	<b>Artajasa</b> Artajasa	
<u>Philippines</u>		
<b>DragonPay</b> dragonpay		
<u>Thailand</u>		
<b>Tesco Lotus via OMISE</b> OMISE_TL	<b>Big Central</b> BigC	
<u>Taiwan</u>		
<b>FamilyMart 全家便利商店</b> FAMILYMART		

\* Channel in yellow text means obsoleted



Merchant is strongly recommended to implement IPN(instant payment notification) in order to acknowledge(**ACK**) on the receiving of payment status from the payment gateway. There are 2 ways to implement IPN. Please refer to **IPN** section for details.

## Examples of Payment Endpoint

You may use the sample for all 3 endpoints, i.e. return URL, notification URL, and callback URL by making little modification based on your own requirements.

### Sample return URL script for PHP

```
<?php

$key = "xxxxxxxxxxxx"; //Replace xxxxxxxxxxxx with Secret_Key

/*****
*Don't change below parameters
*****/
$tranID = $_POST['tranID'];
$orderid = $_POST['orderid'];
$status = $_POST['status'];
$domain = $_POST['domain'];
$amount = $_POST['amount'];
$currency = $_POST['currency'];
$appcode = $_POST['appcode'];
$paydate = $_POST['paydate'];
$key = $_POST['key'];

/*****
* To verify the data integrity sending by PG
*****/
$key0 = md5( $tranID.$orderid.$status.$domain.$amount.$currency );
$key1 = md5( $paydate.$domain.$key0.$appcode.$key );

if( $key != $key1 ) $status= -1; // Invalid transaction.
// Merchant might issue a requery to PG to double check payment status

if ( $status == "00" ) {
    if ( check_cart_amt($orderid, $amount) ) {
        /*** NOTE : this is a user-defined function which should be prepared by merchant ***/
        // action to change cart status or to accept order
        // you can also do further checking on the paydate as well
        // write your script here .....
    }
} else {
    // failure action. Write your script here .....
    // Merchant might send query to PG using Merchant requery
    // to double check payment status for that particular order.
}

// Merchant is recommended to implement IPN once received the payment status
// regardless the status to acknowledge the PG

?>
```

## Sample return URL script for ASP/ASP.NET

```
<!--#include file="md5.asp"--> 'For ASP Developer
<!--#include file="md5.aspx"--> 'For ASP.NET Developer

<%
' md5.asp/md5.aspx is a 3rd party developed md5 solution for ASP/ASP.NET user
' You could get the md5.asp/md5.aspx from support-sa@razer.com
' Some variables below are coming from POST method

dim key0, key1, tranID, orderid, status, merchantID, amount, currency, paydate, appcode, skey

tranID   = Request.Form("tranID")
orderid  = Request.Form("orderid")
status   = Request.Form("status")
merchantID = Request.Form("domain")
amount   = Request.Form("amount")
currency = Request.Form("currency")
paydate  = Request.Form("paydate")
appcode  = Request.Form("appcode")
skey     = Request.Form("skey")
key0     = md5( tranID & orderid & status & domain & amount & currency )
key1     = md5( paydate & merchantID & key0 & appcode & "xxxxxxxxxxxx" )
'Replace xxxxxxxxxxxx with Secret_Key

' invalid transaction if the key is different. Merchant might issue a requery to PG to double check payment status

If skey <> key1 then
    status= -1
End if

If status = "00" then
    ' checking the validity of cart amount & orderid.
    ' if the verification test passed then can update the order status to paid.
    ' you can also do further checking on the paydate as well
Else
    ' failure action
    ' Merchant might send query to PG using merchant requery
    ' to double check payment status for that particular order.
End if

' Merchant is to implement IPN to ack on receiving of payment status
' regardless the payment status

%>
```

## Payment Status Notification (Merchant Webhook or the 3 Endpoints)

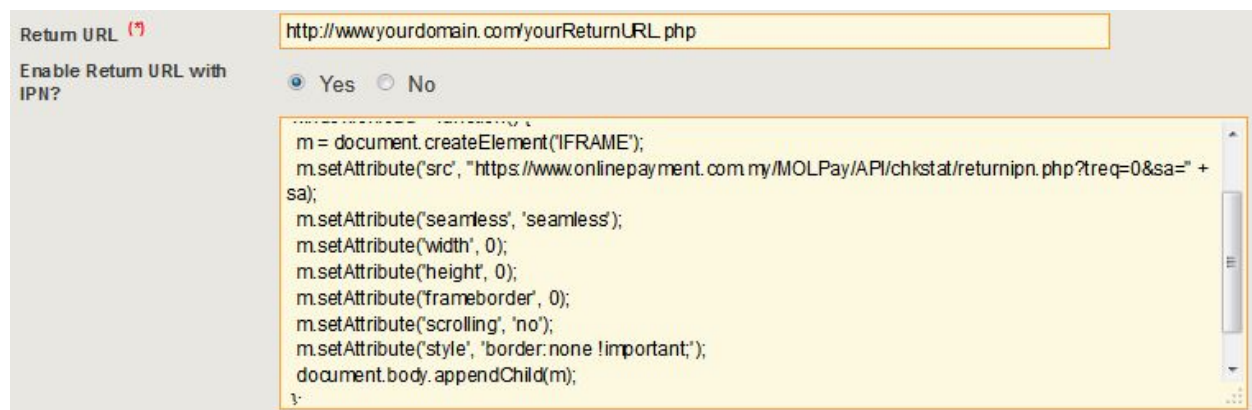
**WARNING:** Please note that multiple payment notifications (either from **return URL**, **notification URL** or **callback URL**) for single transaction is possible but this does not mean that the buyer has paid twice or multiple times.

### Return URL with IPN (Instant Payment Notification)

For normal payment flow, buyer browser is being redirected to hosted payment page, financial institution or channel page(if any), and then return to merchant website or system. User might close the browser any time throughout the payment process, even the payment is completed, successfully or failed. Other possible reason that rarely happens is the network connectivity issue. As a result, payment gateway is unable to update merchant system on the payment status. Therefore, merchant is recommended to implement IPN to acknowledge(ACK) upon the receiving of status from gateway. Otherwise callback worker will resend the payment status within a time interval.

Implementation:

**Step 1:** Logon to merchant admin, choose “Yes” to “Enable Return URL with IPN”, as shown:-



The screenshot shows a merchant admin interface for configuring the Return URL. It includes a text input field for the Return URL, a radio button selection for 'Enable Return URL with IPN', and a text area for pasting JavaScript code.

Return URL (?)

Enable Return URL with IPN? ☒ Yes ☐ No

```
m = document.createElement("IFRAME");
m.setAttribute("src", "https://www.onlinepayment.com.my/MOLPay/API/chkstat/returnipn.php?trq=0&sa=" +
sa);
m.setAttribute("seamless", 'seamless');
m.setAttribute("width", 0);
m.setAttribute("height", 0);
m.setAttribute("frameborder", 0);
m.setAttribute("scrolling", 'no');
m.setAttribute("style", 'border: none !important;');
document.body.appendChild(m);
```

**Step 2:** There are 2 approaches to ack on receiving of payment status.

1. Simple front-end snippet: copy the Javascript (JS) code from merchant admin and paste it on merchant receipt page (which shows payment success/failed), preferable in the HTML header, before `</head>` tag.

2. Advanced back-end scripting: merchant is to echo back all the POST variables with one additional variable, i.e. “*treq*” with value 1. PHP sample code is provided below.

**URL:** <https://www.onlinepayment.com.my/MOLPay/API/chkstat/returnipn.php>

**Step 3:** Merchant to prepare a **notification URL** and **callback URL** script, which is similar to return URL script but serves at the backend, in order to receive consequent payment notification in case merchant system misses the first notification attempt from payment gateway.

## Example of back-end IPN script for PHP (combined with return URL script)

```
<?php

$key = "xxxxxxxxxx"; //Replace xxxxxxxxxx with Secret_Key

$_POST[treq] = 1; // Additional parameter for IPN

// Value always 1. Do not change this value.
$tranID = $_POST['tranID'];
$orderid = $_POST['orderid'];
$status = $_POST['status'];
$domain = $_POST['domain'];
$amount = $_POST['amount'];
$currency = $_POST['currency'];
$appcode = $_POST['appcode'];
$paydate = $_POST['paydate'];
$key = $_POST['key'];

/*****
* Snippet code in purple color is the enhancement required
* by merchant to add into their return script in order to
* implement backend acknowledge method for IPN
*****/
while ( list($k,$v) = each($_POST) ) {
    $postData[] = $k."=".$v;
}
$postdata = implode("&", $postData);
$url = "https://www.onlinepayment.com.my/MOLPay/API/chkstat/returnipn.php";
$ch = curl_init();
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $postData);
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_HEADER, 1);
curl_setopt($ch, CURLOPT_HEADER_OUT, TRUE);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
curl_setopt($ch, CURLOPT_SSLVERSION, 6); // use only TLSv1.2
$result = curl_exec( $ch );
curl_close( $ch );

/*****
* To verify the data integrity sending by PG
*****/
$key0 = md5( $tranID.$orderid.$status.$domain.$amount.$currency );
$key1 = md5( $paydate.$domain.$key0.$appcode.$key );
if( $key != $key1 ) $status = -1; // Invalid transaction
if ( $status == "00" ) {
    if ( check_cart_amt($orderid, $amount) ) {
        // write your script here .....
    }
} else {
    // failure action
}

?>
```

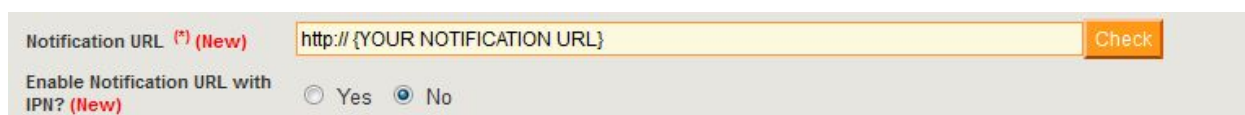
## Notification URL with IPN

Direct payment status notification is a back-end instant update mechanism that send over the payment status notification from gateway directly to merchant server or system, without relying on any user agent such as web browser or mobile application.

Notification URL script is similar to return URL script but no output is required for front-end user interface. This is also known as background URL.

Implementation:

**Step 1:** Logon to merchant admin and go to merchant profile, fill in the Notification URL. Notification URL with IPN can be activated for better merchant-payment system communication. Choose “Yes” to “Enable Notification URL with IPN”, as shown:-



The screenshot shows a form with two main sections. The first section is labeled 'Notification URL (\*) (New)' and contains a text input field with the placeholder 'http:// {YOUR NOTIFICATION URL}' and an orange 'Check' button to its right. The second section is labeled 'Enable Notification URL with IPN? (New)' and contains two radio buttons: 'Yes' and 'No'. The 'No' radio button is selected.

**Step 2:** Merchant to prepare a notification URL script, which is similar to return URL script.

**Step 3:** If IPN is disable please ignore this step. Merchant have to echo back all the POST variables with one additional variable, i.e. “*trreq*” with value 1. PHP sample code is provided below.

**URL:** <https://www.onlinepayment.com.my/MOLPay/API/chkstat/returnipn.php>

## Notification Parameters (via POST method)

Variable / Parameter	Type Format / Max Length	Description / Example
nbcdb	1 digit numeric	Always equal to <b>2</b> , which indicates this is a notification from PG
amount	2 decimal points numeric value	The total amount paid or to be paid for Razer Cash payment request
orderid	alphanumeric, 32 characters	Invoice or order number from merchant system
tranID	integer, 10 digits	Unique transaction ID for tracking purpose
domain	alphanumeric, 32 chars	Merchant ID in PG system
status	2-digit numeric value	00 for Successful payment 11 for failure 22 for pending
appcode	alphanumeric, 16 chars	Bank approval code. Mandatory for card payment. Certain channel returns empty value.

error_code	alphanumeric	Refer to the Error Codes section.
error_desc	text	Error message or description.
skey	32 chars hexadecimal string	This is the data integrity protection hash string. Refer <i>skey</i> section for details.
currency	2 or 3 chars (ISO-4217) currency code	Default currency is MYR (indicating Malaysia Ringgit) for Malaysia channels
channel	predefined string in system	Channel references for merchant system
paydate	Date/Time( YYYY-MM-DD HH:mm:ss)	Date/Time of the transaction.
extraP*	optional (on request)	<p>JSON encoded string or array</p> <p>token: 16-digit card token for merchant to store for recurring MIT (merchant initiated transaction)</p> <p>fraudscreen: 1-digit integer, i.e. 1=Unknown, 2=Passed, 3=Alert, 4=Suspicious, 5=Fraud</p> <p>ccbrand: Visa, MasterCard, AMEX</p> <p>cclast4: Last 4-digit of card number</p> <p>cctype: Credit, Debit, Prepaid</p>

## Example of Notification URL with IPN script for PHP

```
<?php

$key = "xxxxxxxxxx"; //Replace xxxxxxxxxx with Secret_Key

$_POST[treq] = 1; // Additional parameter for IPN. Value always set to 1.

/*****
*Don't change below parameters
*****/
$nbcb = $_POST['nbcb'];
$tranID = $_POST['tranID'];
$orderid = $_POST['orderid'];
$status = $_POST['status'];
$domain = $_POST['domain'];
$amount = $_POST['amount'];
$currency = $_POST['currency'];
$appcode = $_POST['appcode'];
$paydate = $_POST['paydate'];
$key = $_POST['key'];

/*****
* Snippet code in purple color is the enhancement required
* by merchant to add into their notification script in order to
* implement backend acknowledge method for IPN
*****/
while ( list($k,$v) = each($_POST) ) {
    $postData[] = $k."=".$v;
}
$postdata = implode("&", $postData);
$url = "https://www.onlinepayment.com.my/MOLPay/API/chkstat/returnipn.php";
$ch = curl_init();
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $postData);
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_HEADER, 1);
curl_setopt($ch, CURLOPT_HEADER_OUT, TRUE);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
curl_setopt($ch, CURLOPT_SSLVERSION, 6); // use only TLSv1.2
$result = curl_exec($ch);
curl_close($ch);

/*****
* To verify the data integrity sending by PG
*****/
$key0 = md5($tranID.$orderid.$status.$domain.$amount.$currency);
$key1 = md5($paydate.$domain.$key0.$appcode.$key);
if( $key != $key1 ) $status = -1; // Invalid transaction
if ( $status == "00" ) {
    if ( check_cart_amt($orderid, $amount) ) {
        // write your script here .....
    }
} else {
    // failure action
}
```

>

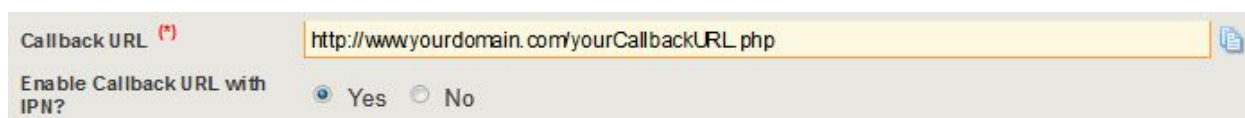


## Callback URL with IPN

Callback mechanism is a back-end activity that initiated by payment gateway to notify merchant system once there are changes on any payment status.

Callback URL script is similar to return URL script. Both handling the payment status result; Unlike the return URL script, callback URL script is to handle defer status update or all other non-realtime payment status update, such as Razer Cash payment.

Merchant must fill the Callback URL in merchant admin in order to get those non-realtime status update from the gateway. Callback URL IPN can be activated for better merchant-payment system communication. Instead of returning all parameters from the gateway, merchant's callback script just need to echo "**CBTOKEN:MPSTATOK**", in plaintext, without double quotes or any HTML tags.



The screenshot shows a form titled "Callback URL (\*)". It contains a text input field with the URL "http://www.yourdomain.com/yourCallbackURL.php". Below the input field, there is a label "Enable Callback URL with IPN?" followed by two radio buttons: "Yes" (which is selected) and "No".

Merchant can actually use the same script for both callback URL & return URL, because it is only one extra parameter in callback request, in addition to return URL parameters, which is "*nbc*". Besides, an additional status code, 22 has been introduced for "pending" transaction status.

Once there is status changed event, payment gateway will POST the following parameters to merchant callback URL. If payment gateway could not get the actual ACK message from merchant, it will retry for maximum 3 times, of 15 minutes interval for each attempt.

### Callback Parameters (via POST method)

Variable / Parameter	Type Format / Max Length	Description / Example
nbc	1 digit numeric	Always equal to 1, which indicates this is a callback notification from PG
amount	2 decimal points numeric value	The total amount paid or to be paid for Razer Cash payment request
orderid	alphanumeric, 32 characters	Invoice or order number from merchant system.
tranID	integer, 10 digits	Unique transaction ID for tracking purpose
domain	alphanumeric, 32 chars	Merchant ID in PG system
status	2-digit numeric value	00 for Successful payment 11 for failure 22 for pending
appcode	alphanumeric, 16 chars	Bank approval code. Mandatory for card payment. Certain channel returns empty value

error_code	alphanumeric	Refer to the Error Codes section
error_desc	text	Error message or description
skey	32 chars hexadecimal string	This is the data integrity protection hash string. Refer skey section for details
currency	2 or 3 chars (ISO-4217) currency code	Default currency is MYR (indicating Malaysia Ringgit) for Malaysia channels
channel	predefined string in system	Channel references for merchant system
paydate	Date/Time( YYYY-MM-DD HH:mm:ss)	Date/Time of the transaction
extraP*	optional (on request)	<p>JSON encoded string or array</p> <p>token: 16-digit card token or Direct Debit token for merchant to store for recurring MIT (merchant initiated transaction)</p> <p>fraudscreen: 1-digit integer, i.e. 1=Unknown, 2=Passed, 3=Alert, 4=Suspicious, 5=Fraud</p> <p>ccbrand: Visa, MasterCard, AMEX</p> <p>cclast4: Last 4-digit of card number</p> <p>cctype: Credit, Debit, Prepaid</p>

## Example of callback URL script for PHP

```
<?php

$key = "xxxxxxxxxx"; //Replace xxxxxxxxxxxx with Secret_Key

$nbcb      =   $_POST['nbcb'];
$tranID    =   $_POST['tranID'];
$orderid    =   $_POST['orderid'];
$status    =   $_POST['status'];
$domain    =   $_POST['domain'];
$amount    =   $_POST['amount'];
$currency  =   $_POST['currency'];
$appcode   =   $_POST['appcode'];
$paydate   =   $_POST['paydate'];
$key       =   $_POST['key'];

/*****
 * To verify the data integrity sending by PG
 *****/
$key0 = md5( $tranID.$orderid.$status.$domain.$amount.$currency );
$key1 = md5( $paydate.$domain.$key0.$appcode.$key );
if( $key != $key1 ) $status= -1; // Invalid transaction

if ( $status == "00" ) {
    if ( check_cart_amt($orderid, $amount) ) {
        // write your script here .....
    }
} else {
    // failure action
    // write your script here .....
}

if ( $nbcb==1 ) {
    //callback IPN feedback to notified PG
    echo "CBTOKEN:MPSTATOK"; exit;
}else{
    //normal IPN and redirection
}
?>
```

# TL;DR?

## SUMMARY ON INTEGRATION, 3 ENDPOINTS AND IPN

### INTEGRATION

- 1) Hosted payment page or normal integration: HTTP POST/GET with all required parameters, either to default payment channel page or to specific channel page, merchant can allow or disallow the buyer to change payment method
- 2) Seamless Integration: popup page or lightbox on merchant checkout flow, as simple as modifying the JavaScript snippet that could be found on GitHub
- 3) Mobile xdk: support variety of mobile development platforms and available on GitHub
- 4) Direct Server API: server-to-server request that allows merchant to handle all the UI/UX with higher flexibility and controllability
- 5) Recurring API: server-to-server request that allows merchant initiated transaction to debit buyer account anytime

### 3 ENDPOINTS for payment response notification

- 1) Return URL: realtime web browser or frontend direction endpoint for hosted page, seamless integration, and shopping cart module
- 2) Notification URL: real time server-to-server or backend endpoint for all kind of integrations
- 3) Callback URL: defer update or callback endpoint on non-real time payment such as Razer Cash

### IPN

- 1) **Frontend** IPN, applicable to return URL endpoint only: using JavaScript, you may copy from the snippet code from your merchant portal, note that **trreq=0**
- 2) **Backend** IPN for return URL and notification URL: post back all parameters with additional parameter **trreq=1** to **returnipn.php**
- 3) IPN for **callback** URL: just echo "**CBTOKEN:MPSTATOK**", without the quote("")

# Comparison Chart

For payment integration, you need to decide which approach fits your business requirements

Integration approach	Hosted payment page	Seamless integration	Inpage checkout	Mobile xdk	Direct server API + CSE	Recurring API
Browser dependency	Yes, popup and redirection	Yes, popup and redirection	Yes, iframe, popup and redirection	Smartphone built-in browser	Depends	No
UI/UX	Moderate	Better	Better	Best for mobile	Handled by merchant	Handled by merchant
Time to market (man-day)	1-7	2-14	2-14	2-14	5-60	10-90
Suitable for	Fast and easy deployment	Better UX	Better UX	In-app purchase	Better UI & UX	Subscription or billing
Mobile readiness	RWD	RWD	RWD	Native / hybrid	Backend only	Backend only
PCI-DSS compliant	PG	PG	PG	PG	PG / Merchant if handling PAN	PG / Merchant if handling PAN
Availability on github	20/22 of the cart modules	6/22 of the cart modules	Yes	12 dev-tools supported	nil	nil
Available endpoints	all 3 endpoints	all 3 endpoints	all 3 endpoints	notification & callback URL	all 3 endpoints	notification & callback URL
Fraud screening	By PG	By PG	By PG	By PG	By merchant	By merchant

For result handling, you need 2 or 3 endpoints(webhook) to capture the payment response

Response endpoint	Return URL	Notification URL	Callback URL
Browser dependency	Yes	No	No
Payment type response	Realtime payment	Realtime payment	Cash payment / defer update
Reliability	Low	High	High
Security	Low if payment verification is not enabled	High	High
IPN implementation	Frontend: easier to implement using JS snippet with req=0  Backend: POST back all values + req=1 to returnipn.php, more reliable	Backend: POST back all values + req=1 to returnipn.php	Backend: echo "CBTOKEN: MPSTATOK"

# MERCHANT REQUEST APIs

PG has prepared plenty of merchant tools for merchant to initiate status requery and cancellation/void of transaction. However, merchant is not allowed to send in requests too frequent (maximum 1 query every 5 seconds). Massive incoming query will lead to IP blocking without prior notice. There will be validity duration to initiate status requery API call, generally within **1 hour** after the initiated time for online payment and the due time for offline payment.

The back-end services available are:-

1. Direct Status Requery - merchant send status query to processing bank directly
2. Indirect Status Requery - merchant send status query to PG system
3. Daily Transaction Report (Reconciliation) - list all transactions of a specific date
4. Settlement Report (Reconciliation) - settlement details
5. Capture Request - to capture any pre-auth or authorized transaction
6. Reversal Request - merchant to cancel or void a transaction or refund of payment
7. Partial Refund - for merchant who requires partial refund only
8. Partial Refund Status Inquiry by Txn ID
9. Channel Status API
10. Card BIN information API
11. Foreign Exchange Rate API
12. Void Pending-Cash API
13. Recurring Plans API

## Notes:

W.e.f 19th October 2017, MOLPay will migrate all none payment-flow related service APIs to a new FQDN: <https://api.molpay.com> from the original <https://www.onlinepayment.com.my>

W.e.f 1st November 2017, MOLPay will shutdown all none payment-flow related service APIs on <https://www.onlinepayment.com.my>

For sandbox environment, use <https://sandbox.molpay.com>

## Direct Status Requery

This will trigger a query to the payment channel or bank status server and there are cases that bank status server is not in-sync with its payment server that might give different result, that leads to a defer update and will trigger a callback from PG server, once the status is synced and changed.

### Request

URL: <https://api.molpay.com/MOLPay/API/gate-query/index.php>

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
amount	2 decimal points numeric value	The payment amount
txID	integer, 10 digits	Unique transaction ID for tracking purpose.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer, obsoleted in new API sets	0 = plain text result (default) 1 = result via POST method

### Response

Variable / Parameter	Type Format / Max Length	Description / Example
Amount	2 decimal points numeric value	The payment amount
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
Channel	alphanumeric, 100 chars	Payment via Channel
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only
ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only

## Formula of skey & VrfKey

```
skey =md5( txID & domain & verify_key & amount )  
VrfKey=md5( Amount & secret_key & Domain & TranID & StatCode )
```

## Example of Direct Status Requery for PHP

```
<?php  
  
$skey = md5($txID . $domain. "xxxxxxxxxx" . $amount);  
//Replace xxxxxxxxxx with Verify Key  
  
echo "<a  
href='https://api.molpay.com/MOLPay/API/gate-query/index.php?amount=3899&txID=65234&domain=shopA&skey=e1c4c60c9  
9116fffc3ce77bd5fd0f7b1'>  
Check payment status for tran ID 65234 </a>";  
  
?>
```

## Example of response

type=0 (default output, plain text with linebreaks)	type=1 (POST result sent to URL)
StatCode=00 StatName=captured TranID=65234 Amount=3899.00 Domain=shopA Channel=fpx VrfKey=456cf69e5bddfe8ed47371096 Currency=MYR ErrorCode= ErrorDesc=	\$_POST [StatCode] => "00"; \$_POST [StatName] => "captured"; \$_POST [TranID] => "65234"; \$_POST [Amount] => "3899.00"; \$_POST [Domain] => "shopA"; \$_POST [Channel] => "fpx"; \$_POST [VrfKey:] => "456cf69e5bddfe8ed47371096"; \$_POST [Currency] => "MYR"; \$_POST [ErrorCode] => ""; \$_POST [ErrorDesc] => "";



## Indirect Status Requery

There are several types of status requery on PG system:-

1. Query by unique transaction ID (recommended)
2. Query by order ID & get latest matched result (single output)
3. Query by order ID & get all matched results (batch output)
4. Query by multiple order ID (batch output)
5. Query by multiple transaction ID (batch output)

For bulk requery, best practice is to utilize the 4th or 5th APIs once every half an hour for unknown status transactions only

### 1. Query by unique transaction ID

#### Request

URL: [https://api.molpay.com/MOLPay/q\\_by\\_tid.php](https://api.molpay.com/MOLPay/q_by_tid.php)

Method: POST or GET

\* Request & Response parameters are same as Direct Status Requery but the format and parameters order of the responses are slightly different.

#### Example of response

type=0 (default output, plain text with linebreaks)	type=1 (POST result sent to URL)
StatCode: 00 StatName: captured TranID: 10645406 Amount: 138.99 Domain: ShopB VrfKey: 9862acf1099b625c00b225887e715861 Channel: credit OrderID: ABC123 Currency: MYR ErrorCode: ErrorDesc:	\$_POST [StatCode] => "00"; \$_POST [StatName] => "captured"; \$_POST [TranID] => "10565234"; \$_POST [Amount] => "3899.00"; \$_POST [Domain] => "shopC"; \$_POST [VrfKey:] => "456cf69e5bddfe8ed47371096"; \$_POST [Channel] => "credit"; \$_POST [OrderID] => "ABC123"; \$_POST [Currency] => "MYR"; \$_POST [ErrorCode] => ""; \$_POST [ErrorDesc] => "";

## 2. Query by order ID (single output)

### Request

URL: [https://api.molpay.com/MOLPay/query/q\\_by\\_oid.php](https://api.molpay.com/MOLPay/query/q_by_oid.php)

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
amount	2 decimal points numeric value	The payment amount
oid	alphanumeric, 32 chars	Merchant order ID, which might be duplicated.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

### Response

Variable / Parameter	Type Format / Max Length	Description / Example
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
OrderID	alphanumeric, 32 chars	Invoice or order number from merchant system.
Amount	2 decimal points numeric value	The payment amount
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date/time
BillingName	alphanumeric, 128 chars	Buyer full name
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
Channel	alphanumeric, 100 chars	Payment via channel
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only

ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX
cclast4	optional, 4-digit numeric with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

#### Formula of skey & VrfKey

skey =md5( olD & domain & verify\_key & amount )  
VrfKey=md5( Amount & secret\_key & Domain & OrderID & StatCode )

### 3. Query by order ID (batch output)

#### Request

URL: [https://api.molpay.com/MOLPay/query/q\\_oid\\_batch.php](https://api.molpay.com/MOLPay/query/q_oid_batch.php)

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
oid	alphanumeric, 32 chars	Merchant order ID, which might be duplicated.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
format	optional, 1-digit integer, for type=1 only	0 = result string with delimiter (   ) 1 = result in array
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

#### Response

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
Amount	2 decimal points numeric value	The payment amount
BillingName	alphanumeric, 128 chars	Buyer full name
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only
ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX

cclast4	optional, 4-digit numeric, with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

## Formula of skey

skey =md5( oID & domain & verify\_key )

## Example of response

type=0, plain text output, newline with single or two TAB character(s)	<table><tr><th>TranID</th><th>BillingDate</th><th>StatCode</th><th>StatName</th><th>Amount</th><th>BillingName</th><th>Currency</th><th>ErrorCode</th><th>ErrorDesc</th></tr><tr><td>418607</td><td>2009-11-26</td><td>22</td><td>pending</td><td>25.00</td><td>Lenka</td><td>MYR</td><td></td><td></td></tr><tr><td>418603</td><td>2009-11-26</td><td>00</td><td>captured</td><td>125.10</td><td>Mika</td><td>MYR</td><td></td><td></td></tr><tr><td>418583</td><td>2009-11-26</td><td>00</td><td>captured</td><td>71.10</td><td>Ciara</td><td>MYR</td><td></td><td></td></tr></table>	TranID	BillingDate	StatCode	StatName	Amount	BillingName	Currency	ErrorCode	ErrorDesc	418607	2009-11-26	22	pending	25.00	Lenka	MYR			418603	2009-11-26	00	captured	125.10	Mika	MYR			418583	2009-11-26	00	captured	71.10	Ciara	MYR		
TranID	BillingDate	StatCode	StatName	Amount	BillingName	Currency	ErrorCode	ErrorDesc																													
418607	2009-11-26	22	pending	25.00	Lenka	MYR																															
418603	2009-11-26	00	captured	125.10	Mika	MYR																															
418583	2009-11-26	00	captured	71.10	Ciara	MYR																															
type=1, format=0, POST variables with delimiter " "	<pre>\$_POST[TranID] = "418607 418603 418583"; \$_POST[BillingDate] = "2009-11-26 2009-11-26 2009-11-26"; \$_POST[StatCode] = "22 00 00"; \$_POST[StatName] = "pending captured captured"; \$_POST[Amount] = "25.00 125.10 71.10"; \$_POST[BillingName] = "Lenka Mika Ciara"; \$_POST[Currency] = "MYR MYR MYR"; \$_POST[ErrorCode] = ""; \$_POST[ErrorDesc] = "";</pre>																																				
type=1, format=1, POST variables in array	<pre>\$_POST[0][TranID] = "418607"; \$_POST[0][BillingDate] = "2009-11-26"; \$_POST[0][StatCode] = "22"; \$_POST[0][StatName] = "pending"; \$_POST[0][Amount] = "25.00"; \$_POST[0][BillingName] = "Lenka"; \$_POST[0][Currency] = "MYR"; \$_POST[0][ErrorCode] = ""; \$_POST[0][ErrorDesc] = ""; \$_POST[1][TranID] = "418603"; \$_POST[1][BillingDate] = "2009-11-26"; \$_POST[1][StatCode] = "00"; \$_POST[1][StatName] = "captured"; \$_POST[1][Amount] = "125.10"; \$_POST[1][BillingName] = "Mika"; \$_POST[1][Currency] = "MYR"; \$_POST[1][ErrorCode] = ""; \$_POST[1][ErrorDesc] = ""; \$_POST[2][TranID] = "418583"; \$_POST[2][BillingDate] = "2009-11-26"; \$_POST[2][StatCode] = "00"; \$_POST[2][StatName] = "captured"; \$_POST[2][Amount] = "71.10"; \$_POST[2][BillingName] = "Ciara"; \$_POST[2][Currency] = "MYR"; \$_POST[2][ErrorCode] = ""; \$_POST[2][ErrorDesc] = "";</pre>																																				

#### 4. Query by multiple order ID (batch output)

##### Request

URL: [https://api.molpay.com/MOLPay/query/q\\_by\\_oids.php](https://api.molpay.com/MOLPay/query/q_by_oids.php)

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
oids	alphanumeric, up to 100 orders	Merchant order ID, must be URLEncoded.
delimiter	single character, default is " "	Avoid using any symbol that might exist in order ID, and also any of these: ",%,*,<,>?,\,\$,&,=
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
format	optional, 1-digit integer, apply for type=1 only	result string with TAB-newline (default) 0 = result string with delimiter ( ) 1 = result in array
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

##### Response

Variable / Parameter	Type Format / Max Length	Description / Example
OrderID	alphanumeric, 32 chars	Merchant order ID for tracking purpose.
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
Amount	2 decimal points numeric value	The payment amount
BillingName	alphanumeric, 128 chars	Buyer full name
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
Channel	alphanumeric, 100 chars	Payment via channel
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR

ErrorCode	alphanumeric	Error code defined by channel for failed transactions only
ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX
cclast4	optional, 4-digit numeric, with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

#### Formula of skey and VrfKey

skey =md5( domain & oIDS & verify\_key )

VrfKey=md5( Amount & secret\_key & Domain & OrderID & StatCode )

## Example of response

type=1 format=1	<pre> [oid1] =&gt; Array (     [TranID] =&gt; 9994238     [BillingDate] =&gt; 2016-11-28 16:32:08     [StatCode] =&gt; 00     [StatName] =&gt; captured     [Amount] =&gt; 30.00     [BillingName] =&gt; kimyoon     [VrfKey] =&gt; 311d72c16e0d3b3fc7994ae93467a2d9     [Channel] =&gt; mb2u     [Currency] =&gt; MYR     [ErrorCode] =&gt;     [ErrorDesc] =&gt; ) [oid2] =&gt; Array (     [TranID] =&gt; 10004613     [BillingDate] =&gt; 2016-11-29 09:56:02     [StatCode] =&gt; 11     [StatName] =&gt; failed     [Amount] =&gt; 58.60     [BillingName] =&gt; MohdAli     [VrfKey] =&gt; f9f06b47e23410e624df5e272accb27dc     [Channel] =&gt; fpx     [Currency] =&gt; MYR     [ErrorCode] =&gt;     [ErrorDesc] =&gt; ) ...  [oidN] =&gt; Array (     [TranID] =&gt; -     [BillingDate] =&gt; -     [StatCode] =&gt; -     [StatName] =&gt; -     [Amount] =&gt; -     [BillingName] =&gt; -     [VrfKey] =&gt; -     [Channel] =&gt; -     [Currency] =&gt; -     [ErrorCode] =&gt; -     [ErrorDesc] =&gt; - ) </pre>
type=1 format=0 delimiter=	<pre> [OrderID] =&gt; oid1 oid2 ... oidN [TranID] =&gt; 9994238 10004613 ... - [BillingDate] =&gt; 2016-11-28 16:32:08 2016-11-29 09:56:02 ... - [StatCode] =&gt; 00 11 ... - [StatName] =&gt; captured failed ... - [Amount] =&gt; 30.00 58.60 ... - [BillingName] =&gt; Nurbaizura KUMARASAN ... - [VrfKey] =&gt; 311d72c16e0d3b3fc7994ae93467a2d9 f9f06b47e23410e624df5e272accb27dc ... - [Channel] =&gt; mb2u fpx ... - [Currency] =&gt; MYR MYR ... - [ErrorCode] =&gt;   ... - [ErrorDesc] =&gt;   ... - </pre>



## 5. Query by multiple transaction ID (batch output)

### Request

URL: [https://api.molpay.com/MOLPay/query/q\\_by\\_tids.php](https://api.molpay.com/MOLPay/query/q_by_tids.php)

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
tIDs	concatenated transaction ID with “ ” up to 100 items	A group of transaction ID, must be URLencoded
domain	alphanumeric, 32 chars	Merchant ID in PG system
skey	32 chars hexadecimal string	This is the data integrity protection hash string
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
format	optional, 1-digit integer, apply for type=1 only	0 = result string with delimiter (   ) 1 = result in array
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

### Response

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Amount	2 decimal points numeric value	The payment amount
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date
BillingName	alphanumeric, 128 chars	Buyer full name
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
OrderID	alphanumeric, 32 chars	Merchant order ID for tracking purpose.
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only

ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX
cclast4	optional, 4-digit numeric, with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

### Formula of skey and VrfKey

$skey = md5(\text{domain} \& \text{tIDS} \& \text{verify\_key})$   
 $VrfKey = md5(\text{Amount} \& \text{secret\_key} \& \text{Domain} \& \text{TranID} \& \text{StatCode})$

## Daily Transaction Report (Reconciliation)

PG Daily Transaction Report provides merchant end-of-day (EoD) reconciliation or to verify all transactions for a specific date.

### Request

URL: <https://api.molpay.com/MOLPay/API/PSQ/psq-daily.php>

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
merchantID	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
rdate	date (YYYY-MM-DD)	The date of transactions to query
status	optional, alphanumeric, 32 chars	00 - success 11 - failed 22 - pending  Combine with delimiter " " for multiple status or left empty for all status.

### Formula of skey

skey =md5( rdate & merchantID & secret\_key )

### Response

(plain text with newline & TAB characters)

Variable / Parameter	Type Format / Max Length	Description / Example
BillingDate	date/time (YYYY-MM-DD HH:mm:ss)	Transaction date/time
OrderID	alphanumeric, 32 chars	Invoice or order number from merchant system.
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Channel	Predefined string in PG system	Channel references for merchant system.
Amount	2 decimal points numeric value	The payment amount
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	word	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
BillingName	alphanumeric, 128 chars	Buyer full name

## Settlement Report (Reconciliation)

PG Settlement Report provides merchant the fund transfer or balance clearance and settlement reconciliation for a specific date.

Version 1.0 and 2.0 will be obsoleted and the following is the latest specification of version 3.0. Version 3.0 will include refund or chargeback records that are excluded from the settlement batch.

### Request

URL: <https://api.molpay.com/MOLPay/API/settlement/report.php>

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
version	Integer, Default value is 1.0.	Indicate version of the API. Current version is 3.0. If not specify, version 1.0 will be use.
merchant_id	alphanumeric, 32 chars	Merchant ID in PG system
token	32 chars hexadecimal string	This is the data integrity protection hash string
date	date (YYYY-MM-DD)	The date of settlement to query
format	json, xml, csv	
download	optional, set to "y" for download mode	Download option is only available for CSV format

### Formula of token

token =md5( merchantID & secret\_key & date )

### Response

Value	Type Format / Max Length	Description / Example
RecordIdentifier	1 character	H: Header
SettlementCurrency	3 chars ISO-4217 currency code	Default is MYR, Settlement currency
SettlementNetAmount	numeric without decimal and comma	total settlement amount, 5331674 is equal to 53,316.74 in that currency
SettlementCommissionAmount	numeric without decimal and comma	total settlement fees, sum of the transfer fees and 3rd party costs
NumberOfTransactions	numeric	total transaction number of settled transaction or D data type in the report, not including R type
BatchReferenceNumber	alphanumeric	Reference number of this settlement

SettlementDate	YYYYMMDD	The settlement date
SettlementGSTAmount	numeric without decimal and comma	Sum of settled transactions GST amount
BankAccount	alphanumeric	Bank swift code and bank account number
RefundNetAmount	numeric without decimal and comma	Sum of the refund and chargeback amount
RefundGSTAmount	numeric without decimal and comma	Sum of the refund and chargeback GST amount

Variable / Parameter	Type Format / Max Length	Description / Example
RecordIdentifier	1 character	D: Data or the content R: Refund / Chargeback Data
MerchantId	alphanumeric	Merchant ID in PG system
OrderId	alphanumeric	Order references for merchant system
Channel	alphanumeric	Payment channel (or method)
AcquirerReference	numeric	PG transaction ID
RefundFees	numeric without decimal and comma	PG refund fees.
TransactionNetAmount	numeric without decimal and comma	amount after deducting MDR/fees
TransactionCommissionAmount	numeric without decimal and comma	MDR+transaction fee+commission+taxes
TransactionDate	YYYYMMDD	The date of transaction been created
TransactionTime	HHmmss	The time of transaction been requested/created
TransactionGrossAmount	numeric without decimal and comma	Payment amount
TransactionCurrency	3 chars ISO-4217 currency code	The currency sent in by merchant or most of the time, paid by the buyer
TransactionGST	numeric without decimal and comma	Malaysia Goods & Services Tax
SettlementNetAmountInProcessingCurrency	numeric without decimal and comma	Settlement amount based on transaction currency after deduct MDR, fees or commissions, except GST
SettlementNetAmount	numeric without decimal and comma	Settlement amount based on settlement currency after deduct MDR, fees or commissions, except GST
SettlementCurrency	3 chars ISO-4217 currency code	MYR, SGD, USD, ...
Forex	numeric without 8 digit decimal and comma	Forex rate for multi currency transaction. Eg: 1.00000000 = 100000000
Status	alphanumeric	The transaction status. Usually SETTLED for D, REFUND for R

### Error (always in JSON format)

Variable / Parameter	Description / Example
success	false
token	invalid token
date	invalid date format, eg. yyyy-mm-dd

### Example (in JSON format)

*The following example might not be accurate and please only take it as a format reference*

```
[
  {
    "RecordIdentifier": "H",
    "SettlementCurrency": "MYR",
    "SettlementNetAmount": "1033293",
    "SettlementCommissionAmount": "2439",
    "NumberOfTransactions": 15,
    "BatchReferenceNumber": "20170817-613",
    "SettlementDate": "20170817",
    "SettlementGSTAmount": "2239",
    "BankAccount": "(MALAYAN BANKING BHD) MerchantID SDN BHD 5144 8457 3110",
    "RefundNetAmount": "179741",
    "RefundGSTAmount": "334"
  },
  {
    "RecordIdentifier": "D",
    "MerchantId": "merchantIDSB",
    "OrderId": "OP-02559",
    "Channel": "maybank2u",
    "AcquirerReference": "18300981",
    "RefundFees": "0",
    "TransactionNetAmount": "9732",
    "TransactionCommissionAmount": "178",
    "TransactionDate": "20170807",
    "TransactionTime": "112250",
    "TransactionGrossAmount": "9900",
    "TransactionCurrency": "MYR",
    "TransactionGST": "010",
    "SettlementNetAmountInProcessingCurrency": "9722",
    "SettlementNetAmount": "000",
    "SettlementCurrency": "MYR",
    "Forex": null,
    "Status": "SETTLED"
  },
  {
    "RecordIdentifier": "D",
    "MerchantId": "merchantIDSB",
    "OrderId": "163138",
    "Channel": "credit",
    "AcquirerReference": "18310736",
    "RefundFees": "0",
    "TransactionNetAmount": "294521",
    "TransactionCommissionAmount": "9656",
    "TransactionDate": "20170807",
    "TransactionTime": "133001",
    "TransactionGrossAmount": "303630",
    "TransactionCurrency": "MYR",
  }
]
```

```
"TransactionGST": "547",
"SettlementNetAmountInProcessingCurrency": "293974",
"SettlementNetAmount": "000",
"SettlementCurrency": "MYR",
"Forex": null,
"Status": "SETTLED"
},
```

... segment removed ...

```
{
"RecordIdentifier": "D",
"MerchantId": "merchantIDSB",
"OrderId": "79126387162378123",
"Channel": "credit",
"AcquirerReference": "18516590",
"RefundFees": "0",
"TransactionNetAmount": "153745",
"TransactionCommissionAmount": "5040",
"TransactionDate": "20170810",
"TransactionTime": "120057",
"TransactionGrossAmount": "158500",
"TransactionCurrency": "MYR",
"TransactionGST": "285",
"SettlementNetAmountInProcessingCurrency": "153460",
"SettlementNetAmount": "000",
"SettlementCurrency": "MYR",
"Forex": null,
"Status": "SETTLED"
},
{
"RecordIdentifier": "D",
"MerchantId": "merchantIDSB",
"OrderId": "31283671293",
"Channel": "credit",
"AcquirerReference": "18532262",
"RefundFees": "0",
"TransactionNetAmount": "9700",
"TransactionCommissionAmount": "318",
"TransactionDate": "20170810",
"TransactionTime": "155958",
"TransactionGrossAmount": "10000",
"TransactionCurrency": "MYR",
"TransactionGST": "018",
"SettlementNetAmountInProcessingCurrency": "9682",
"SettlementNetAmount": "000",
"SettlementCurrency": "MYR",
"Forex": null,
"Status": "SETTLED"
},
```

```
{
"RecordIdentifier": "R",
"MerchantId": "merchantIDSB",
"OrderId": "54671293761293",
"Channel": "credit",
"AcquirerReference": "13251784",
"RefundFees": 0,
"TransactionNetAmount": "-179741",
```

```

"TransactionCommissionAmount": "000",
"TransactionDate": "20170503",
"TransactionTime": "155958",
"TransactionGrossAmount": "-185300",
"TransactionCurrency": "MYR",
"TransactionGST": "-018",
"SettlementNetAmountInProcessingCurrency": "-179741",
"SettlementNetAmount": "000",
"SettlementCurrency": "MYR",
"Forex": null,
"Status": "REFUND"
}

```

]

## Unsettled/Refunded Transaction Report (Exclusion from settlement)

Any unsettled transaction that has been cancelled within a given settlement batch will be retrievable using this API

### Request

URL: [https://api.molpay.com/MOLPay/API/settlement/report\\_refund.php](https://api.molpay.com/MOLPay/API/settlement/report_refund.php)

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
merchant_id	alphanumeric, 32 chars	Merchant ID in PG system
token	32 chars hexadecimal string	This is the data integrity protection hash string
date	date (YYYY-MM-DD)	The date of settlement to query
format	json, xml, csv	
download	optional, set to "y" if	Download option is only available for CSV format

### Formula of token

token =md5( merchantID & secret\_key & date )

### Response

Value	Type Format / Max Length	Description / Example
H	1 character	Header
Settlement currency	3 chars ISO-4217 currency code	Default is MYR
total settlement amount	numeric without decimal and comma	5331674 is equal to 53,316.74 in that currency



total settlement fees	numeric without decimal and comma	Sum of the transfer fees and 3rd party costs
total refund transaction number	numeric	Total number of refund transactions
batch reference number	alphanumeric	Reference number of this settlement
date	YYYYMMDD	The settlement date
bank account	alphanumeric	Bank swift code and bank account number
Start date	YYYY-MM-DD	The settlement period start date
End date	YYYY-MM-DD	The settlement period end date

Variable / Parameter	Type Format / Max Length	Description / Example
R	1 character	Refund Data
merchant ID	alphanumeric	Merchant ID in PG system
Order ID	alphanumeric	Order references for merchant system
Channel / transaction ID	alpha-numeric / numeric	Payment channel / PG transaction ID
Transaction net amount	Numeric without decimal and comma	This should equal the original transaction amount minus the fee or commission.
Transaction commission amount	Numeric without decimal and comma	The MDR or transaction fee or commission
Forex Rate	0	0
Settlement amount	0	0
Settlement currency	0	0
Transaction date	YYMMDD	The date of settlement
Transaction gross amount	Numeric without decimal and comma	Original transaction amount.
Transaction currency	3 chars ISO-4217 currency code	Default is MYR
Status	alpha-numeric	The transaction status. Usually REFUND.

#### Error (always in JSON format)

Variable / Parameter	Description / Example
success	false
token	invalid token
date	invalid date format, eg. yyyy-mm-dd

## Capture Request (For pre-auth or authorized transaction)

Merchant who uses preauth or authorize payment mode may capture the transaction at a later stage by using this API

### Request

URL: <https://api.molpay.com/MOLPay/API/capstxn/index.php>

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system.
tranID	integer, 10 digits	Unique PG transaction ID.
amount	2 decimal points numeric value	The total amount paid or to be paid.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.

### Response (JSON Format)

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 10 digits	Unique transaction ID for tracking purposes.
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
StatCode	2-digit numeric	00 = Success 11 = Failure 12 = Invalid or unmatched security hash string 13 = Not a credit card transaction 15 = Requested day is on settlement day 16 = Forbidden transaction 17 = Transaction not found 18 = Missing required parameter 19 = Domain not found 20 = Temporary out of services.
StatDate	date (YYYY-MM-DD)	Response date & time

### Formula of skey & VrfKey

skey =md5( txnID & amount & domain & verify\_key )  
VrfKey=md5( secret\_key & Domain & TranID & StatCode )

## Reversal Request

Merchant can request a reversal of transaction via original payment method for an “authorized” card, and “captured” card (including recurring payment), M2U, CIMB Clicks, Hong Leong Connect, Razer Pay, Alipay-Spot, POS Terminal, Boost and WeChat Pay payment.

For most of the payment channel above, the transaction can be “void” immediately on the same day before settlement (card payment at around 10pm and other channels are 11:59pm local time). For payment channel that accepts refund request, the transaction that happens within 180 days will be refunded within 7-14 days after the request sent.

Please note that this API is to send refund request, but not getting the status of the refund process. All successful request shall be executed and if there is any exceptional case, PG support team will contact the merchant to resolve the issue.

### Request

URL: <https://api.molpay.com/MOLPay/API/refundAPI/refund.php>

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
txnID	integer, 10 digits	Unique transaction ID for tracking purpose.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
key	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method

### Response

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
StatCode	2-digit numeric	00 = Success (will proceed the request) 11 = Failure 12 = Invalid or unmatched security hash string 13 = Not a refundable transaction 14 = Transaction date more than 180 days 15 = Requested day is on settlement day 16 = Forbidden transaction 17 = Transaction not found 18 = Duplicate partial refund request

		19 = Merchant not found 20 = Missing required parameter
StatDate	date (YYYY-MM-DD HH:mm:ss)	Response date & time

### Formula of skey & VrfKey

skey =md5( txnID & domain & secret\_key)  
VrfKey=md5( secret\_key & Domain & TranID & StatCode )

### Reversal Reference Table

Payment Channel	Support Void / Refund / Both	Cut-off time for VOID request	Processing method	Credit to buyer within
MYR card payment	Both	10pm GMT+8	manual	14-business day
Maybank2u	Refund	-	manual	7-business day
CIMB Clicks	Refund	-	manual	7-business day
Hong Leong Connect	Refund	-	manual	7-business day
Razer Pay	Both	11:59pm GMT+8	auto	2-business day
Alipay Spot	Void	11:59pm GMT+8	auto	2-business day
Wechat Pay	Void	11:59pm GMT+8	auto	2-business day
Boost	Void	11:59pm GMT+8	auto	2-business day

## Advanced Full/Partial Refund

Merchant can request full/partial refund for a “captured” or “settled” transaction regardless of the payment method. The request can be sent within 180 days from the transaction creation date and the refund process will take about 7-14 days after the request sent.

Merchant can enable the “**Refund Portal**” so that buyer who needs refund could easily provide the bank account details to shorten the refund lead time.

### Request

URL: <https://api.molpay.com/MOLPay/API/refundAPI/index.php>

Method: POST or GET

Field Name	Data Type(Size)	M/O	Description
RefundType	a{1}	M	P - Partial Refund
MerchantID	an{1..32}	M	Merchant ID provided by PG
RefID	ans{1..100}	M	Unique reference ID for tracking purpose sent by merchant
TxnID	n{1..10}	M	PG Transaction ID
Amount	n{10,2}	M	eg. '5.00' Amount to be refunded
BankCode	an{8}	C	Applicable for Online Banking and Physical Payment transaction only. (Refer to predefined bank lists)
BankCountry	a{2}	C	Applicable for Online Banking and Physical Payment transaction only. Two letters country ISO code. (Default value: MY)
BeneficiaryName	as{1..100}	C	Applicable for Online Banking and Physical Payment transaction only.
BeneficiaryAccNo	ans{1..100}	C	Applicable for Online Banking and Physical Payment transaction only.
Signature	an{32}	M	This is data integrity protection hash string.
mdr_flag	n{1}	O	This is to include or exclude MDR refund to buyer if the amount is same as bill amount. Available value is as below: 0 - Include MDR/Full Refund (Default) 1 - Exclude/Reserved MDR
notify_url	as	O	This is the URL for merchant to receive refund status (same response as this API), either 00 (Success) or 11 (Rejected) with slightly different Signature

Signature = md5( RefundType . MerchantID . RefID . TxnID . Amount . **secret\_key** )

## Response

PG responds JSON format to merchant upon a successful request (positive result)

No	Field Name	Data Type(Size)	M/O	Description
1	RefundType	a{1}	M	Echo of merchant request
2	MerchantID	an{1..32}	M	Echo of merchant request
3	RefID	ans{1..100}	M	Echo of merchant request
4	RefundID	n{1..11}	M	Refund ID provided by PG
5	TxnID	n{1..10}	M	Echo of merchant request
6	Amount	n{10,2}	M	Echo of merchant request
7	Status	n{2} [00, 11, 22]	M	22 for 'Pending' (Upon Request) 11 for 'Rejected' (via notify_url) 00 for 'Success' (Refunded, via notify_url)
8	Signature	a{32}	M	This is data integrity protection hash string.
9	reason	ans{1..255}	O	Reason for rejected status

Signature = md5( RefundType . MerchantID . RefID . RefundID . TxnID . Amount . Status . **secret\_key** )

PG will respond the following in JSON format once error occurs (negative result)

No	Field Name	Data Type(Size)	Description
1	error_code	an{5}	Refer to Appendix C
2	error_desc	ans{1..255}	Refer to Appendix C

## Partial Refund Status Inquiry by TxnID/RefID

Merchant is able to do status inquiry for refund transaction.

### Request Type 1

URL: [https://api.molpay.com/MOLPay/API/refundAPI/q\\_by\\_txn.php](https://api.molpay.com/MOLPay/API/refundAPI/q_by_txn.php)

Method: POST or GET

No	Field Name	Data Type(Size)	M/O	Description
1	TxnID	n{1..10}	M	PG Transaction ID
2	MerchantID	a{1..32}	M	Merchant ID provided by PG
3	Signature	n{1..32}	M	This is data integrity protection hash string.

Signature = md5( TxnID . MerchantID . verify\_key )

### Request Type 2

URL: [https://api.molpay.com/MOLPay/API/refundAPI/q\\_by\\_refID.php](https://api.molpay.com/MOLPay/API/refundAPI/q_by_refID.php)

Method: POST or GET

No	Field Name	Data Type(Size)	M/O	Description
1	RefID	ans{1..100}	M	Unique tracking/references ID from merchant
2	MerchantID	a{1..32}	M	Merchant ID provided by PG
3	Signature	n{1..32}	M	This is data integrity protection hash string.

Signature = md5( RefID . MerchantID . verify\_key )

### Response

PG responds JSON format to merchant upon a successful request (positive result)

No	Field Name	Data Type(Size)	M/O	Description
1	TxnID	n{1..10}	M	Echo of merchant request
2	RefID	ans{1..100}	M	Unique tracking/references ID from merchant
3	RefundID	n{1..11}	M	Refund ID provided by PG
4	Status	a{8}	M	'pending', 'rejected', 'success'
5	LastUpdate	ans{1..255}	M	Last update timestamp.

## Appendix for Partial Refund and Partial Refund Status Inquiry

### Appendix A : Data Type Details

No	Code	Description
1	a	Letters, A-Za-z
2	n	Numbers, 0-9
3	s	Symbols, .: ?*,!&_-
4	{x}	Fixed length x
5	{y..x}	Length range: y – x
6	{y,x}	Number range: 0-9. 0-9

### Appendix B : M/O Details

No	Code	Description
1	M	Mandatory field.
2	O	Optional field.
3	C	Conditional field.

### Appendix C : Error Code & Description in JSON

No	error_code	error_desc
1	PR001	Refund Type not found.
2	PR002	MerchantID field is mandatory.
3	PR003	RefID field is mandatory.
4	PR004	TxnID field is mandatory
5	PR005	Amount field is mandatory.
6	PR006	Signature field is mandatory
7	PR007	Merchant ID not found.
8	PR008	Invalid Signature.
9	PR009	Txn ID not found.
10	PR010	Transaction is not settled yet.
11	PR011	Exceed refund amount for this transaction.
12	PR012	Bank information is not applicable for credit channel transaction.
13	PR013	BankCode not found in our database, please contact support.



14	PR014	Bank information is mandatory for non-credit channel transaction.
15	PR015	Server is busy, try again later.
16	PR016	Duplicate RefID found, please provide a unique RefID.
17	PR017	Refund request for transaction that is out of the allowed period.
18	PR018	BeneficiaryName cannot contain non-alphanumeric characters.
19	INQ001	TxnID field is mandatory.
20	INQ002	MerchantID field is mandatory.
21	INQ003	Signature field is mandatory.
22	INQ004	Merchant ID not found.
23	INQ005	Invalid Signature.
24	INQ006	Unable to find refund transaction.
25	INQ011	RefID field is mandatory.

## Static QR-Code Generator

For merchant to generate static QR code of e-wallet

### Request

URL: <https://api.molpay.com/MOLPay/API/staticqr/index.php>

Method: POST or GET

No	Field Name	Data Type(Size)	M/O	Description
1	merchantID	Alphanumeric, 32 chars	M	Merchant ID provided by PG
2	channel	Alphanumeric, 32 chars	M	Channel requested: <ul style="list-style-type: none"><li>AlipaySQR - Alipay Static QR</li><li>WeChatPaySQR - WeChat Pay Static QR</li></ul>
3	orderid	Alphanumeric, 32 chars	M	Items ID. E.g: S001
4	currency	Alphabet, 3 chars	M	ISO-4217 currency code.
5	amount	Numeric	M	The total amount to be paid in one purchase order. 2 decimal points and comma( , ) is not allowed.
6	bill_name	Alphanumeric, 128 chars	M	Items name.
7	bill_desc	Text	M	Items description.
3	checksum	Alphanumeric, 32 chars	M	This is request integrity protection hash string.

Checksum = md5( merchantID . channel . orderid . currency . amount . verify\_key )

### Response

PG responds JSON format to merchant upon a successful request

No	Field Name	Data Type(Size)	M/O	Description
1	status	Boolean	M	true/false
2	qrcode_data	URL	M	QR-Code data. Available for status true.
3	qrcode_link	URL	M	Link to view QR-Code image. Available for status true.
4	error_code	Alphanumeric	M	Error code when status is false.
5	error_mesg	Text	M	Error description when status is false.

error_code	error_mesg
SQ001	Missing required parameter (<FieldName>).
SQ002	System is busy now, temporary out of services. Please try again later.
SQ003	Merchant info not found.

SQ005	Invalid checksum value.
SQ004	Currency not supported.
SQ006	Your account doesn't subscribe to this channel. Please contact our support teams to enable this.

## Channel Status API

In order to know the healthiness of payment channels, this API allows system-wide or merchant-only successful rate (OK rate) of a channel in real time for frequently used payment channel. Merchant can always check the latest 1 hour system-wide OK rate of the channel (card payment and internet banking) and a few window sizes, i.e. the latest 1, 3, 6, and 12 hours for merchant-only OK rate for all channels, including Razer Cash. Keep in mind that Razer Cash is not a real-time payment channel and sampling is based on payment request time and not the time that payment is made.

This API gives worst case scenario result, please DO NOT approach RMS support team if your number of transactions of a specific payment channel is less than 30 or if there are multiple failure attempt from the same buyer. Try to optimize your OK rate (success rate or SR) if yours are far below the system-wide level. Frequent query on merchant-only OK rate might slow down your payment processing as well.

Prompt alert or warning instead of turning off the channel at 0% or low system-wide OK rate is always a best practice.

### Request

URL: <https://api.molpay.com/MOLPay/API/chkstat/OK-rate.php>

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system.
reqTime	YYYYMMDDHHmmss	Request date & time, e.g. 20161202153423
reqType	Global, Merchant	Global: system-wide OK rate Merchant: merchant only OK rate
skey	32 chars hexadecimal string	For merchant access verification purpose
duration	1, 3, 6, 12 (for reqType=merchant only)	Latest sampling window size in <b>HOURL</b> for merchant-only request. If the total sampling number is less than 30, it has less reference value due to the low statistical significance and n/a will be given

### Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
StatTime	YYYYMMDDHHmmss	Status date & time, e.g. 20161202153435
OK-rate	{ "Channel 1" : OK_rate1, "Channel 2" : OK_rate2,	Channel name with successful rate measure at the StatTime. Please refer to "Direct Server API" or "Seamless Integration" for channel name.

	<pre> ... } </pre>	<pre> {   "credit": 75,   "maybank2u": 70,   "cimbclicks": 65,   "fpx": 56,   "hlb": 60,   "rhb": 52,   ...   "fpx_pbb": 45,   "fpx_bimb": 80, } </pre> <p>where the OK rate is an integer range from 0-100. Channel might be added/removed without prior notice.</p>
--	--------------------	---

### Formula of skey

skey = md5( domain & secret\_key & reqTime & reqType )

## Card BIN information API

To retrieve the card BIN information such as card type and the issuer information.

### Request

URL: [https://api.molpay.com/MOLPay/query/q\\_BINinfo.php](https://api.molpay.com/MOLPay/query/q_BINinfo.php)

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	For merchant access verification purpose
BIN	6-digit numeric	First 6-digit number of the PAN, e.g. 519603

### Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
card_brand	VISA / MASTERCARD / AMEX / UP / JCB / DC	MASTERCARD
card_issuer	ans{100}	CIMB BANK BERHAD
debit_credit	DEBIT / CREDIT / PREPAID / N/A	CREDIT
card_type	ans{100}	STANDARD other example like GOLD, PLATINUM, CLASSIC, co-brand name is available
card_country	ISO-3166-1 alpha-2	MY
islamic_card	Y / N	Y

PG will respond the following in JSON format once error occurs (negative result)

No	Field Name	Data Type(Size)	Description
1	error_code	an{7}	Refer to Appendix A
2	error_desc	ans{1..255}	Refer to Appendix A

### Formula of skey

skey = md5( domain & secret\_key & BIN )

**Appendix A : Error Code & Description in JSON**

No	error_code	error_desc
1	QBIN001	Missing required Parameter.
2	QBIN002	Merchant info not found.
3	QBIN003	Invalid skey.
4	QBIN004	Card BIN info not found.
5	QBIN005	System is busy now, temporary out of services. Please try again later.

## Foreign Exchange Rate API

In order to know the current foreign exchange (FX) of each supported currency, this API allows merchant to know the current exchange rate for each supported currency.

### Request

URL: [https://api.molpay.com/MOLPay/query/q\\_fx\\_rate.php](https://api.molpay.com/MOLPay/query/q_fx_rate.php)

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system.
reqtime	YYYYMMDD	Request date & time, e.g. 20161202
source	optional, 1-digit integer	Predefined value as below: 1. BNM 2. Maybank
skey	32 chars hexadecimal string	For merchant access verification purpose

### Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
base	3 chars ISO-4217 currency code	MYR
reqtime	YYYYMMDD	20170128
rate	float(2,4) in array	1.2888
[currency] under "rate"	3 chars ISO-4217 currency code	Currency to compared with MYR (USD,SGD,etc)

### Formula of skey

skey = md5( domain & secret\_key & reqTime )

Will list out the list of currency, exchange rate compared to 1 MYR with rate expiration date/time.

```
{
  "base": "MYR"
  "reqtime": "20170128"
  "rate":{
    "USD":{"units":1,"rates":0.24425989252565},
    ...
  }
}
```



## Void Pending-Cash API

For merchant to cancel and void the cash payment request order, before getting paid or the expiry time, and force-to-expired.

### Request

URL: <https://api.molpay.com/MOLPay/API/VoidPendingCash/index.php>

Method: POST or GET

No	Field Name	Data Type(Size)	M/O	Description
1	tranID	n{1..10}	M	PG Transaction ID
2	amount	n{10,2}	M	The total amount to be paid in one purchase order. 2 decimal points, comma( , ) is not allowed.
3	merchantID	an{1..32}	M	Merchant ID provided by PG
4	checksum	an{32}	M	This is request integrity protection hash string.

Checksum= md5( tranID . amount . merchantID . verify\_key )

### Response

PG responds JSON format to merchant upon a successful request

No	Field Name	Data Type(Size)	M/O	Description
1	StatCode	n{2}	M	00 = Success (voided) 11 = Missing required parameter (<FieldName>). 12 = Merchant info not found 13 = Invalid checksum value. 14 = Transaction not found 15 = Transaction not Pending  99 = System is busy now, temporary out of services. Please try again later.
2	tranID	n{1..10}	O	PG Transaction ID
3	orderid	an{1..32}	O	Merchant order
4	amount	n{10,2}	O	The total amount to be paid in one purchase order
5	merchantID	an{1..32}	O	Merchant ID provided by PG
6	channel	an{1..32}	O	Channel references for merchant system.

## Recurring Plans API

For merchants to retrieve recurring plans info which were created from merchant portal.

### Request

URL: [https://api.molpay.com/MOLPay/API/Recurring/get\\_plans.php](https://api.molpay.com/MOLPay/API/Recurring/get_plans.php)

Method: POST or GET

No	Field Name	Data Type(Size)	M/O	Description
1	domain	an{1..32}	M	Merchant ID provided by PG
2	charge_on_endofmonth	a{1}	O	Predefined value as below: 1. Y 2. N
3	period	a{3..8}	O	Predefined value as below: 1. month 2. week 3. day 4. year 5. quarter 6. halfyear 7. bimonth 8. biweek
4	cycle_term	n{1..10}	O	Billing cycle (E.g. '6', '12', '24')
5	status	a{2..3}	O	Predefined value as below: 1. On 2. Off 3. NA
6	skey	an{32}	M	This is request integrity protection hash string.

### Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
planID	n{1..10}	Unique ID for each plan.
plan_name	ans{1..64}	Plan name.
plan_desc	ans{1..255}	Plan description.
amount	n{10,2}	The total amount to be paid in each cycle.
period	a{3..8}	Predefined value as below: 1. month 2. week 3. day 4. year 5. quarter 6. halfyear 7. bimonth 8. biweek

status	a{2..3}	Predefined value as below: 1. On 2. Off 3. NA
cycle_term	n{1..10}	Billing cycle (E.g. '6', '12', '24')
charge_on_endofmonth	a{1}	Predefined value as below: 1. Y 2. N

PG will respond the following in JSON format once error occurs (negative result)

No	Field Name	Data Type(Size)	Description
1	error_code	an{7}	Refer to Appendix A
2	error_desc	ans{1..255}	Refer to Appendix A

### Formula of skey

skey = md5( domain & secret\_key & charge\_on\_endofmonth & period & cycle\_term & status)

### Appendix A : Error Code & Description in JSON

No	error_code	error_mesg
1	PLAN001	Missing required parameter (<FieldName>).
2	PLAN002	Merchant info not found.
3	PLAN003	Invalid skey.
4	PLAN004	Plan not found.
5	PLAN005	System is busy now, temporary out of services. Please try again later.
6	PLAN006	Invalid parameter value (<FieldName>).

# ERROR CODES

## Payment Page

Error Code	Description
P01	Timeout
P02	Your transaction has been denied due to merchant account issue. Payment channel is not available for merchant.
P03	Your payment info format not correct ! Incorrect security hash string, check "vcode".
P04	not allow to process Incomplete buyer information, check bill_name, bill_mobile, bill_email, orderid, faked mobile such as 0123456789 will be blocked as well
P05	Payment gateway not found Invalid payment gateway file or channel code.
P06	System is busy now, temporary out of services. Please try again later.
P07	Access Denied. Requestor not authorize. Merchant URL is not allowed to process payment or not yet registered with PG.
P08	Invalid referral
P09	Duplicate payment is not allowed for this merchant. Payment with same order ID and amount is not allowed to capture twice. Cash payment is not allowed to request twice.
P10	Sorry, Your Credit Card Number or CVV or expiration date is not valid
P11	Amount return from bank not match with PG system.
P12	Signature from bank not match.
P13	Currency not supported.
P14	Transaction amount must more than CUR XXX.XX.
P15	Transaction amount must less than CUR XXX.XX.
P16	Invalid subMerchantID
P17	Unable to retrieve the currency exchange rate
P18	Empty currency rate
P21	Massive errors detected from the same IP address

P22	Massive errors detected from the same IP address
P33	System is busy
P44	Bill description format incorrect
P45	Subtotal amount not match with bill amount
P403	Invalid payment URL
P404	Invalid merchant ID
P990	Sorry, we're not able to process your request now. You may close this window.
P991	System temporary not available due to security issue
P1813	URL is not allowed
R01	Error in payment process due to empty key
T01	Tokenization error
01	Unsuccessful Timeout due to user did not authorize the payment
99	Record not found Timeout due to user did not proceed with the payment
999	Transaction is blocked Due to one of the reason: country restriction, blacklist, exceeds transaction limit

### 3DS Error

Error Code	Description
P3011	This merchant only allows 3DS card
P3089	Invalid 3DS request
P3099	Card not authorized or invalid BIN
MD2310	rejected due to non-3DS (a feature of PG to block non-3DS transaction)

## Payment Status Query (PSQ) Error

Error Code	Description
Q01	Missing Required Parameter ( <i>{parameter_name}</i> ).
Q02	Invalid Parameter Value ( <i>{parameter_name}</i> ).
Q03	System is busy now, temporary out of services. Please try again later.
Q04	Merchant info not found.
Q05	Send result to merchant URL failed.
Q06	Query period expired. No result available for transaction more than 7 days.
Q99	Service blocked due to massive connection. Please send only 1 request every 5 minutes.
Q101 / Q201	Correct skey with invalid data
Q00004 / Q102 / Q202 / Q302 / Q402 / Q502	Incorrect skey
Q203 / Q303	Transaction record not found
Q401	Delimiter error

## Error in Merchant Admin

Error Code	Description
M0001	Inactive merchant account
M0002	Frozen merchant account
M0003	Merchant account has been purged
M0098	Invalid merchant ID
M0099	Unable to process the request
M1007	Invalid bank account number
M1008	Invalid email
M1009	Invalid bank account number & email

## Error Codes in Channel

PG will add a the following prefix to all credit card and certain internet banking error codes

Credit Card (Paymex, MIGS, etc)	CC_
FPX Internet Banking	FPX_

Error Code   Description		
<b>Prepaid/Debit/Credit Card via Paymex (Visa/MasterCard)</b>		
1XX   Invalid input to 3D Secure MPI 2XX   Error related to 3D authentication 3XX   Error related to authorization 4XX   System error or timeout		
Error Code	Description	Action / Remarks
000	Transaction is successful	Merchant to display the confirmation page to cardholder
101	Invalid field passed to 3D Secure MPI	Merchant needs to check error description to find out what is wrong with the field. Authorization/Authentication not carried out.
201	Invalid ACS response format. Transaction is aborted.	Retry the transaction. If error persists, contact the issuing bank.
202	Cardholder failed the 3D authentication, password entered by cardholder is incorrect and transaction is aborted	Merchant to display error page to cardholder
203	3D PaRes has invalid signature. Transaction is aborted	Retry the transaction. If error persists, contact the issuing bank.
300	Transaction not approved	Transaction has failed authorization, e.g. due to insufficient credit, invalid card number, etc. The actual response code provided by acquiring host can be found via the View Transaction History web page available to merchants.
301	Record not found	<ul style="list-style-type: none"> <li>Merchant/User has submitted a transaction with invalid purchase ID</li> <li>Merchant/User tried to reverse a previously declined transaction</li> </ul>
302	Transaction not allowed	<ul style="list-style-type: none"> <li>Purchase ID not unique due to mismatched card number and/or transaction amount</li> <li>System unable to process reversal due to transaction</li> </ul>

		<ul style="list-style-type: none"> <li>has been settled</li> <li>System unable to process reversal due to transaction type is CAPS</li> <li>System unable to process previously voided transaction</li> </ul>
303	Invalid Merchant ID	Not a valid merchant account
304	Transaction blocked by error 901	Merchant to report error to acquiring bank
305	Merchant password is not available	The error code will be returned if merchant password is not available during the VISA 3D authentication,
306	Merchant exceeded count limit	The error code will be returned if the merchant daily transaction count exceeded the predefined daily transaction count limit.
307	Merchant exceeded amount limit	The error code will be returned if the merchant daily transaction amount exceeded the predefined daily transaction amount limit.
308	Merchant exceeded group count limit	The error code will be returned if the merchant transaction exceeded the predefined merchant organization transaction count limit.
309	Merchant exceeded group amount limit	The error code will be returned if the merchant transaction exceeded the predefined merchant organization transaction amount limit.
310	Request signature is not available	The PX_SIG is not append on the PxReq message or PX_SIG is blank
311	Request signature verification failed	The signature of the PxReq is not match with the PxReq message
900	3D Transaction timeout	Timeout of 3D transaction due to late response from Issuer ACS, after the predefined 3D timeout set in the application.
901	System Error	System unable to complete transaction. Merchant to report error to acquiring bank.
902	Time out	Issuing/acquiring host timeout, transaction is not approved
903	Pxtems no response	Transaction has failed to get response from MPI links. The service not available due to: <ul style="list-style-type: none"> <li>Database connection error</li> <li>Network connection error</li> </ul>

#### Host Error

00 | Successful approval/completion or that V.I.P. PIN verification is valid  
 01 | Refer to card issuer  
 02 | Refer to card issuer, special condition  
 03 | Invalid merchant or service provider  
 04 | Pickup card  
 05 | Do not honor / Incorrect CVV or 3D password  
 06 | Error  
 07 | Pickup card, special condition (other than lost/stolen card)  
 10 | Partial Approval-Private label  
 11 | V.I.P. approval  
 12 | Invalid transaction  
 13 | Invalid amount (currency conversion field overflow. Visa Cash-invalid load mount)  
 14 | Invalid account number (no such number)  
 15 | No such issuer  
 19 | Re-enter transaction  
 21 | No action taken (unable to back out prior transaction)  
 25 | Unable to locate record in file, or account number is missing from the



inquiry  
 28 | File is temporarily unavailable  
 41 | Pickup card (lost card)  
 43 | Pickup card (stolen card)  
 51 | Insufficient funds  
 52 | No checking account  
 53 | No savings account  
 54 | Expired card  
 55 | Incorrect PIN (Visa Cash-invalid or missing S1 signature)  
 57 | Transaction not permitted to cardholder (Visa Cash-incorrect routing, not a load request)  
 58 | Transaction not allowed at terminal  
 59 | Suspected fraud"; break;  
 61 | Activity amount limit exceeded  
 62 | Restricted card (for example, in Country Exclusion table)  
 63 | Security violation  
 65 | Activity count limit exceeded  
 75 | Allowable number of PIN-entry tries exceeded  
 76 | Unable to locate previous message (no match on Retrieval Reference number)  
 77 | Previous message located for a repeat or reversal, but repeat or reversal data are in consistent with original message  
 80 | Invalid date (For use in private label card transactions and check acceptance transactions)  
 81 | PIN cryptographic error found (error found by VIC security module during PIN decryption)  
 82 | Incorrect CVV/iCVV  
 83 | Unable to verify PIN  
 85 | No reason to decline a request for account number verification or address verification  
 91 | Issuer unavailable or switch inoperative (STIP not applicable or available for this transaction)  
 92 | Destination cannot be found for routing  
 93 | Transaction cannot be completed; violation of law  
 96 | System malfunction System malfunction or certain field error conditions  
 N0 | Force STIP  
 N3 | Cash service not available  
 N4 | Cash request exceeds issuer limit  
 N7 | Decline for CVV2 failure  
 P2 | Invalid biller information  
 P5 | PIN Change/Unblock request declined  
 P6 | Unsafe PIN

#### **Credit Card via MIGS (Visa/MasterCard)**

0 | Transaction Successful  
 ? | Transaction status is unknown  
 1 | Unknown Error  
 2 | Bank Declined Transaction  
 3 | No Reply from Bank  
 4 | Expired Card  
 5 | Insufficient funds  
 6 | Error Communicating with Bank  
 7 | Payment Server System Error  
 8 | Transaction Type Not Supported  
 9 | Bank declined transaction (Do not contact Bank)  
 A | Transaction Aborted  
 C | Transaction Cancelled  
 D | Deferred transaction has been received and is awaiting processing  
 F | 3D Secure Authentication failed  
 I | Card Security Code verification failed  
 L | Shopping Transaction Locked (Please try the transaction again later)  
 N | Cardholder is not enrolled in Authentication scheme  
 P | Transaction has been received by the Payment Adaptor and is being

processed  
R | Transaction was not processed - Reached limit of retry attempts allowed  
S | Duplicate SessionID (OrderInfo)  
T | Address Verification Failed  
U | Card Security Code Failed  
V | Address Verification and Card Security Code Failed  
  
Y | The cardholder was successfully authenticated.  
E | The cardholder is not enrolled.  
N | The cardholder was not verified.  
U | The cardholder's Issuer was unable to authenticate due to some system error at the Issuer.  
F | There was an error in the format of the request from the merchant.  
A | Authentication of your Merchant ID and Password to the ACS Directory Failed.  
D | Error communicating with the Directory Server.  
C | The card type is not supported for authentication.  
S | The signature on the response received from the Issuer could not be validated.  
P | Error parsing input from Issuer.  
I | Internal Payment Server system error.  
  
05 | Fully Authenticated  
06 | Not authenticated (cardholder not participating), liability shift  
07 | Not authenticated due to a system problem

#### **PayNet FPX Internet Banking**

00 | Approved  
03 | Invalid Merchant  
05 | Seller Account Closed  
09 | Transaction Pending  
12 | Invalid Transaction  
13 | Invalid Amount  
14 | Invalid Buyer Account  
20 | Invalid Response  
30 | Transaction Not Supported For Model Or Format Error  
31 | Invalid Bank  
39 | No Credit Bank  
45 | Duplicate Seller Order Number  
46 | Invalid Seller Exchange Or Seller  
47 | Invalid Currency  
48 | Maximum Transaction Limit Exceeded  
49 | Merchant Specific Limit Exceeded  
50 | Invalid Seller for Merchant Specific Limit  
51 | Insufficient Funds  
53 | No Saving Account  
57 | Transaction Not Permitted  
58 | Transaction to Merchant Not Permitted  
65 | Withdrawal Frequency Exceeded  
70 | Invalid Serial Number  
76 | Transaction Not Found  
77 | Invalid Buyer Name Or Buyer Id  
78 | Decryption Failed  
79 | Host Decline When Down  
80 | Buyer Cancel Transaction  
83 | Invalid Transaction Model  
84 | Invalid Transaction Type  
85 | Internal Error At Bank System  
87 | Debit Failed Exception Handling  
88 | Credit Failed Exception Handling  
89 | Transaction Not Received Exception Handling  
90 | Bank Internet Banking Unavailable  
92 | Invalid Buyer Bank  
96 | System Malfunction

98 | MAC Error  
 99 | Pending for Authorization (Applies to B2B model)  
 BB | Blocked by Bank  
 BC | Transaction Cancelled By Customer  
 DA | Invalid Application Type  
 DB | Invalid Email Format  
 DC | Invalid Maximum Frequency  
 DD | Invalid Frequency Mode  
 DE | Invalid Expiry Date  
 DF | Invalid e-Mandate Buyer Bank ID  
 FE | Internal Error  
 OE | Transaction Rejected As Not In FPX Operating Hours  
 OF | Transaction Timeout  
 SB | Invalid Seller Bank Code  
 XA | Invalid Source IP Address (Applicable for B2B2 model)  
 XB | Invalid Seller Exchange IP  
 XC | Seller Exchange Encryption Error  
 XE | Invalid Message  
 XF | Invalid Number of Orders  
 XI | Invalid Seller Exchange  
 XM | Invalid FPX Transaction Model  
 XN | Transaction Rejected Due To Duplicate Seller Exchange Order Number  
 XO | Duplicate Exchange Order Number  
 XS | Seller Does Not Belong To Exchange  
 XT | Invalid Transaction Type  
 XW | Seller Exchange Date Difference Exceeded  
 1A | Buyer Session Timeout at Internet Banking Login Page  
 1B | Buyer failed to provide the necessary info to login to IB Login page  
 1C | Buyer choose cancel at Login Page  
 1D | Buyer session timeout at Account Selection Page  
 1E | Buyer failed to provide the necessary info at Account Selection Page  
 1F | Buyer choose cancel at Account Selection Page  
 1G | Buyer session timeout at TAC Request Page  
 1H | Buyer failed to provide the necessary info at TAC Request Page  
 1I | Buyer choose cancel at TAC Request Page  
 1J | Buyer session timeout at Confirmation Page  
 1K | Buyer failed to provide the necessary info at Confirmation Page  
 1L | Buyer choose cancel at Confirmation Page  
 1M | Internet Banking Session Timeout  
 2A | Transaction Amount Is Lower Than Minimum Limit

#### Alipay (China)

Error Code	Description
ILLEGAL_SIGN	Illegal signature
ILLEGAL_ARGUMENT	Illegal parameters
HASH_NO_PRIVILEGE	No sufficient rights to complete the query
ILLEGAL_SERVICE	Service Parameter is incorrect
ILLEGAL_PARTNER	Incorrect Partner ID
ILLEGAL_SIGN_TYPE	sign_type parameter is incorrect
FOREX_MERCHANT_NOT_SUPPORT_THIS_CURRENCY	Cannot support this kind of currency
ILLEGAL_SECURITY_PROFILE	Cannot support this kind of encryption
REPEAT_OUT_TRADE_NO	out_trade_no parameter is repeated
ILLEGAL_CURRENCY	Currency parameter is incorrect

ILLEGAL_PARTNER_EXTERFACE	Service is not activated for this account
SYSTEM_EXCEPTION	Contact Alipay technical support for help
ILLEGAL_TIMEOUT_RULE	Timeout_rule parameter is incorrect
ILLEGAL_CHARSET	Illegal charset

<b>BCard</b>  00 Transaction done without error 01 Invalid Card/invalid owner of card 02 Invalid Merchant 03 Invalid PIN 04 Insufficient Point 05 PIN Changed Fail 06 Card Is Blocked 07 Card Is Expired 08 Principal Is Currently A Supplementary 09 Supplementary Is Currently A Principal 10 Supplementary Is Already A Supplementary 11 Merge Own Card 12 Transaction Record Does Not Exists/ Invalid Transaction ID 13 Transaction Already Voided 14 Transaction Already Reversed 15 Transaction record more than one or not found 16 Invalid point/amount/bill no 18 Invalid category 19 Card not registered 20 Duplicate Data 21 Web Service Key already setup 22 Invalid Web Service Key 23 Invalid IC 24 Invalid Name 25 Invalid Operation/adjustment 26 Invalid STAN No/ Offline ID 27 Invalid Staff 29 Invalid Date 98 Unexpected Error 99 Server Exception
<b>CIMB Clicks</b>  00 APPROVED OR COMPLETED 01 REFER TO CARD ISSUER 03 INVALID MERCHANT 04 CAPTURE CARD 05 DO NOT HONOR 12 INVALID TRANSACTION 13 INVALID AMOUNT 14 INVALID CARD NUMBER 15 INVALID ISSUER 30 FORMAT ERROR 41 LOST CARD 43 STOLEN CARD 51 INSUFFICIENT FUNDS 52 NO CURRENT ACCOUNT 53 NO SAVING ACCOUNT 54 EXPIRED CARD 55 INVALID PIN 57 TRANSACTION NOT PERMITTED ISS 58 TRANSACTION NOT PERMITTED ACQ 59 SUSPECTED FRAUD

61 EXCEEDS WITHDRAWAL LIMITS  
 62 RESTRICTED CARD  
 63 SECURITY VIOLATION  
 65 EXCEEDS WITHDRAWAL COUNT LIMIT  
 68 LATE RESPONSE  
 75 ALLOWABLE PIN TRIES EXCEEDED  
 80 FORCE PIN CHANGE  
 81 FORCE PRE-REGISTERED PIN CHANGE  
 91 SWITCH OR ISSUER INOPERATIVE  
 92 UNABLE TO ROUTE TRANSACTION  
 94 DUPLICATE TRANSACTION DETECTED  
 96 SYSTEM ERROR/ MALFUNCTION/VALIDATION ERROR

**Dragonpay (Philippines)**

000 Success  
 101 Invalid payment gateway id  
 102 Incorrect secret key  
 103 Invalid reference number  
 104 Unauthorized access  
 105 Invalid token  
 106 Currency not supported  
 107 Transaction cancelled  
 108 Insufficient funds  
 109 Transaction limit exceeded  
 110 Error in operation  
 111 Invalid parameters  
 201 Invalid Merchant Id  
 202 Invalid Merchant Password

**eNETS Debit (Singapore)**

"For the online transaction status query feature, the response returned to the merchant is in the format of <stage>\_<responsecode>."

Stage Code	Stage Description
000001	Payment Request from eNETS II to Bank Host
000002	Notification from Bank Host to eNETS II
000003	Acknowledgement from eNETS II to Bank Host
000004	Txn End from Bank Host to eNETS II
000005	Payment Request from Merchant to eNETS II
000006	Notification from eNETS II to Merchant
000007	Acknowledgement from Merchant to eNETS II
000008	Txn End from eNETS II to Merchant

Error Code	Description	Action / Remarks
00000	Transaction Successful	Merchant to proceed with fulfillment of order. Note: Full response code should be 000008_000000
20001	Internal System Error	Internal processing errors such as Database unavailability and network problem, etc. If Gateway returns this status for Notification Acknowledgement Message, the Bank will have to initiate a

		reversal of the transaction.
20002	Data Validation Error	Invalid data sent from either party The bank will not proceed with the transaction and will initiate a reversal. It will inform customer the transaction is unsuccessful.
20003	XML Related Error	Either XML parsing error or invalid/missing DTD etc. If Gateway returns this status for Notification Acknowledgement Message, DBS will have to initiate a reversal of the transaction.
20004	Transaction Failed	Transaction is not successful, due to reasons such as customer has entered invalid PIN or insufficient funds, etc. Customer will be notified of the status and be redirected back to the Merchant page.
20005	Communication Errors	Open connection type of errors, broken pipe errors or timeout errors. For notification message, the Bank will retry a configurable number of times.
20006	Digital Certificate Errors	Either party's certificate is either expired or invalid. The Bank/Gateway will not accept the message and if the notification message has already been sent, the Bank will initiate a reversal.
20007	Invalid digital Signature	Verification of either party's signature failed. The Bank/Gateway will not accept the message and if the notification message has already been sent, the Bank will initiate a reversal.
20008	Session Errors	Time out due to customer inactivity. Customer will not be allowed to proceed and will be informed that his/her session has expired. They will then be redirected back to the Merchant site.
20009	Bank API configuration error	Inform Banks to reconfigure API.
20010	Failed to receive acknowledgement from Gateway	Check Gateway.
20011	No match for Gateway public certificate.	Double check Gateway public certificates.
30001	Merchant not found in Merchant_Profile table	No merchant record in the merchant_profile table. Check Gateway database.
30002	Merchant Failure_URL not found in Merchant_Notify_URL table.	Check Gateway database.
30003	Unable to retrieve Merchant's Public key from Merchant_Cert table.	Check Gateway database.
30004	Signature Error	Unable to verify Merchant signature due to different keys used. Merchant to check that the public key sent to the

		Gateway is correct.
30005	Duplicate Transaction	The payment request consists of a Merchant reference code that was used before. Merchant to advise customer to wait before trying again.
30006	Merchant Ref Code generated is not unique.	Merchant to advise customer to wait before trying again.
30007	Unable to insert transaction to MerchantArchive table.	Check gateway database.
30008	Unable to save transaction to ASPArchive table.	Check gateway database.
30009	Unable to insert transaction into Merchant_Txn_Info and Merchant_DD_Txn_Info tables.	Check gateway database.
30010	Unable to insert ASP Txn Time into ASP_Txn_Info table.	Check gateway database.
30011	Unable to generate gateway signature	Merchant should check the public key of Gateway.
30012	Bank ID not found.	Transaction cancelled by user at Gateway's Bank selection page. Merchant can show the same payment page again.
30013	Bank ID not enabled	Fill in the problem log and escalate to eNETS.
30014	Client-side HTTP error when notifying Merchant.	Check Gateway HTTP.
30015	Server-side HTTP error when notifying Merchant.	Check Gateway HTTP.
30016	Merchant Notification Acknowledgement Timeout	Merchant's notification acknowledgement received later than expected turnaround time.
30017	Exchange Rate does not exist (Forex Debit)	Required for currency conversion Contact RBS and/or verify Rate Retrieval from FXmicropay
30018	Invalid Exchange Rate (Forex Debit)	Required for currency conversion Contact RBS and/or verify Rate Retrieval from FXmicropay
30019	System does not allow transaction to be processed	Transaction rejected by switch during initial stage. Possible causes: <ul style="list-style-type: none"> <li>- profile is not active</li> <li>- amount not in range,</li> <li>- invalid merchant</li> <li>- invalid currency</li> <li>- invalid payment mode</li> <li>- invalid bank id</li> </ul>
40001	Unable to retrieve Bank' URL from Bank_Profile table.	Check Gateway database.

40002	Merchant Ref code gotten from Bank is null.	Check that BankMgr is running.
40003	Unable to retrieve Bank's Public key from Database	Check Gateway database.
40004	Unable to verify Bank Signature.	The Gateway will not accept the message. Check with the respective bank.
40005	Unable to save Txn End html to BankArchive table.	Check Gateway database.
40006	Unable to save Txn End html to ASPArchive table.	Check Gateway database.
40007	Unable to insert transaction into Log_Merchant_Trans_End table.	Check Gateway database.
40008	Unable to update status in Merchant_Txn_Info table.	Check Gateway database.
40009	Unable to retrieve Merchant_URL in Merchant_Notify_URL table.	Check Gateway database.
40010	Duplicate Txn End	The payment request consists of a bank reference code that was used before. Bank to advise customer to wait before trying again.
40011	Transaction timeout	The Gateway has no response in specified time. Check Gateway itself and link between Gateway and bank.
40012	Gateway private key has been revoked	
40013	Bank private key has been revoked	
40014	Merchant private key has been revoked.	
40096	System Error	
40097	Transaction Not Found	
40098	Transaction is not in a valid state for this action.	Stage response code does not align with the transaction status. Check gateway database
40099	Transaction Unsuccessful	Transaction cancelled by user at Gateway's Bank selection page. Merchant can show the same payment page again.
50001	Registry Access Error	Fill in the problem log and escalate to eNETS
50002	Database Access Error	Fill in the problem log and escalate to eNETS
50003	Page forward Error	Fill in the problem log and escalate to eNETS
60001	Unable to get the CRL	Fill in the problem log and escalate to eNETS.



60002	Unable to send email to Administrator	Fill in the problem log and escalate to eNETS.
60003	Unable to validate the merchant Certificate	Fill in the problem log and escalate to eNETS.
60004	Unable to update revoked flag in database for Merchants	Check the database.
60005	Unable to validate the Gateway Certificates	Fill in the problem log and escalate to eNETS.
60006	Unable to update revoked flag in database for Gateway	Check the database.
60007	Unable to validate the Bank Certificates	Fill in the problem log and escalate to eNETS.
60008	Unable to update revoked flag in database for Banks	Check the database.
60009	No certificate Authority ID specified	Fill in the problem log and escalate to eNETS.
70001	Unable to connect to the Password Server	Double check password.
70002	Invalid database password	Key into the correct password.
70003	Failed to check connection to the database	Fill in the problem log and escalate to eNETS.
80001	Error loading data into context	Fill in the problem log and escalate to eNETS.
90001	Invalid URL for reloading data into context	Fill in the problem log and escalate to eNETS.
1001	Wrong merchant or customer identification.	SS should check whether this merchant's records are in the database or not.
1002	Wrong bank PIN keyed-in by user.	User should key in again.
1003	Internet Banking transaction failed. Please check with the Bank concerned.	SS should check with the Bank.
1004	Session timeout.	Transaction has taken too long or user is inactive for too long. Customers will not be allowed to proceed and will be informed that their session has expired. They will then be redirected back to the Merchant site.
1005	Database access error.	Check Gateway database.
1006	Error in re-direction of web page.	Check Gateway connection with Banks or Merchants.

1007	Invalid digital signature.	Verification of Gateway signature failed The Gateway will not accept the message.
1008	Network error.	Check Gateway network.
1009	Unable to connect to bank server.	Call respective bank to check his bank.
1010	Error due to user PC shutdown unexpectedly.	
1011	User cancelled transaction before completion.	
1035	Invalid UMID	

#### Gift Card

Error Code	Description
0	Success
1	Failure
2	Invalid ESN
3	Vendor Application Error
4	System Error
5	Card Already Active
6	Card Already Inactive
7	Invalid Request
8	Card Redeemed
9	Invalid Card
10	Card Deactivated
11	Card Suspended
12	Card Activated
13	Invalid Account
14	No Previous Action
15	Card is Active
16	Card is Deactivated
17	Card is Redeemed
18	Card is Not Found
19	Card Locked
20	Delayed Redemption
21	Maximum Transactions Exceeded

22	Invalid XML
23	Network Error
24	Invalid PIN
25	Declined
26	PIN Not Sold
27	PIN Already Sold
28	Card Expired
29	Card Stolen
30	Card Lost
31	Card Damaged
32	Card Pending
33	PIN Redeemed
34	Invalid/Inactive PIN
35	Card In Use
36	Already Reversed
37	Not Reversible
38	Incorrect Card Status
39	Card Not Active
40	Pin Locked
41	Suspected Fraud
42	Format Error
43	Action Not Supported
44	Card Action Suspended
45	Card is Pending
46	Card is Suspended
47	Card is Expired
48	Card is Stolen
49	Card is Lost
50	Card is Damaged
<b>Multi Currency Credit Card (Visa/MasterCard/AMEX)</b>	
<b>Error Code</b>	<b>Description</b>

GeneralFailure	An unspecified error has occurred.
RateOfferNotFound	The RateOfferId did not identify a known, non-expired GoInterpay rate offer. One common reason for this result is that the rate offer has expired.
CurrencyNotFound	The CustomerCurrency did not correspond to a supported GoInterpay currency.
CurrencyMismatch	The CustomerCurrency specified in the request did not match the Currency associated with the specified GoInterpay rate offer.
EmailAddressInvalid	The consumer Email was not supplied or was supplied but was syntactically invalid.
PaymentMethodUnsupported	The PaymentMethod did not specify a supported type, or the merchant does not have access to the specified type for this payment.
CardNameInvalid	The Name in the card details was not valid.
CardNumberInvalid	The Number in the card details was not valid for the specified payment method.
CardYearInvalid	The Year in the card details was not valid.
CardMonthInvalid	The Month in the card details was not valid.
CardExpired	The specified payment card has expired.
CardVerificationCodeInvalid	The VerificationCode in the card details was not valid.
PostalCodeInvalid	The consumer PostalCode was not supplied and is required for the merchant's GoInterpay payment processing option, or was supplied but was syntactically invalid.
CountryInvalid	The consumer Country was not supplied and is required for the merchant's GoInterpay payment processing option, or was supplied but was syntactically invalid.
AmountInvalid	A ConsumerPrice or MerchantPrice did not meet the requirements documented for the request.
AmountLimitExceeded	The request contains an amount that exceeds the amount that GoInterpay is willing to transact in a single transaction for the merchant.
PaymentAuthorizationFailed	The payment details were syntactically correct, but the payment processing provider (or an intermediary) refused to authorize a payment against the payment details for the specified amount. This includes the case of failure to pass payment processor fraud checks. This also includes the failure of the payment processing provider to validate authentication performed by a payment authentication mechanism (via the Merchant).
Blacklisted	The consumer is on the GoInterpay blacklist and GoInterpay has refused to process the order.
FraudSuspected	Based on the parameters of the order, GoInterpay suspects that

	the order is fraudulent and/or represents too high a risk to proceed with processing.
ContractNotFound	The specified ContractId is not associated with the requesting Merchant.
ContractClosed	The specified ContractId is closed and cannot be used to begin new transactions.
ViaAgentInvalid	The ViaAgent flag was specified for a payment method that does not support it.
InstalmentsInvalid	A Number was specified for Instalments which is not between 2 and 12 (inclusive) or instalments are not available for this order.
BirthDateInvalid	The consumer BirthDate was not valid, or is missing but is required for this order.
NationalIdentifierInvalid	The consumer NationalIdentifier was not valid, or is missing but is required for this order.
PhoneInvalid	The consumer Phone was not valid, or is missing but is required for this order.
RegionInvalid	The consumer Region was not valid, or is missing but is required for this order.
<b>Hong Leong Online (HONG LEONG Connect)</b>	
Error Code	Description
0000	Success
0101	Invalid fingerprint hash
0102	Invalid message request
0103	Invalid seller ID
0104	Invalid seller account
0105	Bank host down for maintenance
0106	Invalid login
0107	Invalid mobile number
0108	No debiting account
0109	Exceed transaction limit
0110	Exceed TAC attempts

0111	Purchase item not available
0112	Insufficient fund in buyer account
0113	Invalid account status of buyer or seller account
0114	Transaction rejected by host or reversed payment
0115	Buyer session timeout on Login Page
0116	Buyer cancel payment on Login Page
0117	Buyer session timeout on Account Selection Page
0118	Buyer cancel payment on Account Selection Page
0119	Buyer session timeout on Confirmation Page
0120	Buyer cancel payment on Confirmation Page
0121	Buyer invalid session
0122	Payment not found
0123	Processing transaction
0124	Transaction rejected by reversed payment
0125	Transaction is not allowed
0199	General exception error

**PEX Plus (HONG LEONG PEX+)**

Error Code	Description
00000	Successful.
00149	User has not logged in or session expired.
00341	Your session has timed out due to inactivity. Please login again.
00344	Not a privileged user.
10000	No data found.
44444	Service is not available.
00121	Payment in progress.
00201	QR sales has expired.
00300	QR code has been paired.
30002	Invalid currency.
00102	Invalid amount.

12001	Exceeded daily transaction limit.
00301	Cancellation failed.
00302	Cancellation successful.
01201	Merchant is not valid.
01202	Organisation is not valid.
00903	Batch No. is not valid.
10007	Sorry, we are unable to process your request. Please try again later.
10008	Merchant has been deactivated.
10010	Merchant has not subscribed to PEx+ services.
10011	This transaction has been reversed.
10013	Sorry, you are not allowed to perform this action.
10014	Sorry, this transaction cannot be reversed due to return policy.
PEX+.ERR_PAYMENT.10102	The Payment Transaction is invalid. Please try again. [10102]
PEX+.ERR_PAYMENT.10105	The Payment Transaction is invalid. Please try again. [10105]
PEX+.1000000	No Record Found.
PEX+SD.00412	The transaction amount has exceeded the daily PEx+ transaction limit. [PEX+.0100010]
PEX+SD.00403	Sorry, your transaction has been cancelled. Please perform another PEx+ payment.
PEX+SD.00404	Sorry, your transaction has expired. Please perform another PEx+ payment.
PEX+SD.30002	Currency Code is invalid. Please try again.
PEX+SD.01301	Invalid QR scanning. Please perform another PEx+ payment.
PEX+SD.30001	Record Not Found. Please try again later.
PEX+.0100000	Sorry, we are unable to process your request. Please try again later. [PEX+.0100000]
PEX+.0100011	Your account balance is insufficient for the transaction. [PEX+.0100011]
PEX+_rbs.5	Sorry, we are unable to process your request at this time. Please try again later.
N/A	Invalid QR code. Please try again.
<b>ATM Cash Permata Bank (Indonesia)</b>  000 Success 100 Invalid merchant ID 101 Empty merchant ID 102 Empty amount	

103 Invalid amount (not approved string,dot and comma )  
 104 Empty orderid  
 105 Empty bill\_name  
 106 bill\_email  
 107 Empty bill\_mobile  
 108 Empty bill\_desc  
 109 Empty vcode  
 110 Invalid Authentication Key  
 111 Empty due\_time  
 113 Invalid

#### MOLPoints

Error Code	Error Description
40001	Required parameter is required. / Parameter format is invalid.
40002	Invalid API Version.
40003	Invalid Currency Code or not supported.
40004	Duplicate Reference Id. The transaction was refused as a result of a duplicate Reference Id supplied. Currency Code is not match with previous transaction. The transaction was refused as a result of a duplicate Reference Id supplied. Pin is not match with previous transaction.
40005	Invalid Channel Id.
40006	Invalid Amount.
40007	Invalid PIN.
40008	Invalid Client IP Address.
40009	The transaction was declined by MOL because of possible fraudulent activity.
40013	Payment Amount Exceed channel maximum accepted amount.
40014	Payment Amount less than channel minimum accepted amount.
40101	Invalid Application Code.
40102	Unauthorized Server IP Address.
40103	Invalid Signature.
40104	Channel Id not permitted.
40400	Payment not found.

#### POLi Payment (Australia & New Zealand)

Error Code	Description	Action / Remarks
1001	Invalid Token	The token specified in the request corresponds to a POLi ID that does not exist in the database. There may be an error in the implementation of POLi™. Check your web services.



1002	Illegal Token	The token specified in the request corresponds to a POLi ID that does not belong to the specified merchant. There may be an error in the implementation of POLi™. Check your web services.
1003	Invalid Merchant Code	The merchant code specified in the request does not exist in the database. There may be an error in the implementation of POLi™. Check your web services.
1004	Inactive Merchant	The merchant code specified in the request corresponds to a merchant that is inactive. There may be an error in the implementation of POLi™. Check your web services.
1005	Merchant Not Authenticated	The merchant authentication code supplied is not correct or the authentication type has not been specified in the POLi™ system. There may be an error in the implementation of POLi™. Check your web services.
1006	Deleted Merchant	The merchant code specified in the request corresponds to a merchant that has been deleted. There may be an error in the implementation of POLi™. Check your web services.
1007	Invalid Currency Code	The specified currency code does not exist in the database. There may be an error in the implementation of POLi™. Check your web services.
1008	Invalid Merchant Currency	The specified currency code does not correspond to an active currency for the specified merchant. There may be an error in the implementation of POLi™. Check your web services.
1009	Currency System Limit Exceeded	The payment amount in the specified currency has exceeded the system limit for that currency. Inform the customer that POLi™ applies transaction limits for security reasons and to try another payment limit. Do not specify the limit.
1010	Currency VAR Limit Exceeded	The payment amount in the specified currency has exceeded the VAR limit for that currency. Inform the customer that POLi™ applies transaction limits for security reasons and to try another payment limit. Do not specify the limit.
1011	Currency Merchant Single Transaction Limit Exceeded	The payment amount in the specified currency has exceeded the merchant's single transaction limit for that currency. Inform the customer that POLi™ applies transaction limits for security reasons and to try another payment limit. Do not specify the limit.
1012	Currency Merchant Daily Cumulative Limit Exceeded	The payment amount in the specified currency has exceeded the merchant's daily cumulative limit for that currency.

		Inform the customer that POLi™ applies transaction limits for security reasons and to try another payment limit. Do not specify the limit.
1013	Invalid Merchant Established Date Time	The difference between the specified merchant established time and the system time is more than 24 hours. Check your date and time settings.
1014	Invalid URL Format	The format of the specified URL is invalid. There may be an error in the implementation of POLi™. Check your web services.
1015	Invalid timeout value	The specified timeout value is less than the system minimum timeout value. may be an error in the implementation of POLi™. Check your web services.
1016	The transaction has expired	The transaction being enquired upon has lapsed past the 15min enquiry window. Use the Merchant Console to attain the outcome of this transaction.
1017	Blocked User IP address	The IP address of the user is blocked, restricted or otherwise from a list of known suspect IP addresses. Do not try to initiate a transaction again. NOTE: Strongly recommend that the user not be allowed to complete the transaction using another payment option at the Merchant's discretion.
1018	Invalid IP address format	The IP address is in an invalid format. Try again passing in the correct data.
1019	Invalid IP address	The IP address is invalid. Try again passing in the correct data.
1020	No merchant primary account	The merchant has not set up a primary account to be used. Set up a primary account for the merchant through the console and try initiating the transaction again after the change has been approved.
1021	Invalid Field Characters	The specified field contains invalid characters. There may be an error in the implementation of POLi™. Check your web services.
1022	Mandatory Field	No value is supplied for a mandatory field. There may be an error in the implementation of POLi™. Check your web services.
1023	Invalid Field Length	The specified field has an invalid length. There may be an error in the implementation of POLi™. Check your web services.
1024	Invalid Currency Amount In Field	The specified field contains invalid currency amount. There may be an error in the implementation of POLi™. Check your web services.
1025	Invalid Field Range	The value in the field is out of the allowable range. There may be an error in the implementation of POLi™.

		Check your web services.
1026	Invalid Transaction Status	The transaction has not followed the anticipated transaction status path. NOTE: This error should never be returned to a Merchant. If it does then please inform Centricom.
1027	Invalid Merchant Financial Institution	The Financial Institution Code passed in is not allowed for this merchant. There may be an error in the implementation of POLi™. Check your web services.
1028	Invalid Financial Institution Code	The Financial Institution Code passed in is not valid. There may be an error in the implementation of POLi™. Check your web services.
1029	Inactive Financial Institution	The Financial Institution Code passed in is not currently active. There may be an error in the implementation of POLi™. Check your web services.
1030	Deleted Financial Institution	The Financial Institution Code passed in has been deleted. There may be an error in the implementation of POLi™. Check your web services.
1031	Invalid Financial Institution Vector	The vector for the passed in Financial Institution Code is not available or non-existent. NOTE: This error should never be returned to a Merchant. If it does then please inform Centricom
1032	Invalid Transaction Status Code	The Transaction Status Code passed in is not valid. There may be an error in the implementation of POLi™. Check your web services.
2021	Invalid Payment Amount	The amount being paid is not equal to the transaction amount. The user's locale can affect the way that currency amounts are displayed on the screen. For example, a German locale may display the amount with a comma rather than a decimal point. This affects the way that POLi™ interprets the amount and may result in a failure. Setting the locale to En-AU, En-US, or En-GB will rectify this issue.
8001	Operational Error Without Trace Information	An operational error occurs but there is no trace information available. Perform the web service again.
8002	Operational Error With Trace Information	An operational error occurs and trace information is available. Perform the web service again.
8003	Invalid Field Characters	The specified field contains invalid characters. There may be an error in the implementation of POLi™. Check your web services.
8004	Mandatory Field	No value is supplied for a mandatory field. There may be an error in the implementation of POLi™. Check your web services.

8005	Invalid Field Length	The specified field has an invalid length. There may be an error in the implementation of POLi™. Check your web services.
8006	Invalid Currency Amount In Field	The specified field contains invalid currency amount. There may be an error in the implementation of POLi™. Check your web services.
8007	Invalid Field Range	The value in the field is out of the allowable range. There may be an error in the implementation of POLi™. Check your web services.
11002	Unable to Send Nudge	A nudge sent to the merchant has failed. The Nudge Notification URL may be publicly inaccessible or the destination is down or it takes too long to respond.
12001	Merchant Code Required	Merchant code is empty. Provide a non-empty merchant code.
12002	Merchant Code Length	Merchant code length exceeds maximum number of characters allowed. Provide a Merchant Code with valid length.
12003 or 1003	Invalid Merchant Code	Merchant Code doesn't exist or inactive. Provide a valid Merchant Code.
12004	Authentication Code Required	Authentication Code is empty. Provide a non-empty Authentication Code.
12005	Authentication Code Length	Authentication Code length exceeds maximum number of characters allowed. Provide an Authentication Code with valid length.
12006 or 1005	Invalid Authentication Code	Provide a valid Authentication Code.
12010	Request Type is Required	Request Type is empty. Provide a non-empty Request Type.
12011	Request Type Length	Request Type length exceeds maximum number of characters allowed. Provide a Request Type with a valid length.
12012	Invalid Request Type	Request Type is not valid. It must be either Manual or Email. Provide a Request Type set to either Manual or Email.
12013	Invalid Payment Amount.	Payment Amount is empty. Either it is empty or not a decimal number or decimal precision exceeds 2 or not in the range of 1.00 and 10000.00 Provide a valid Payment Amount.
12014	Payment Reference is required	Payment reference is empty. Provide a non-empty Payment Reference.
12015	Invalid Payment Reference	Payment reference is not valid. Provide a Payment Reference with a valid length.
12016	Invalid Confirmation Email.	Confirmation Email is either empty or not one of the values Yes, No. Provide Confirmation Email with a valid value.

12017	Invalid Customer Reference.	Customer Reference is either empty or not one of the values Yes, No. Provide Customer Reference with a valid value.
12018	Recipient Name is required.	Recipient Name is empty. Provide a non-empty Recipient Name.
12019	Recipient Name Length	Recipient Name length exceeds maximum number of characters allowed. Provide a Recipient Name with a valid length.
12020	Invalid Recipient Name.	Recipient Name is not valid. Check invalid characters in the Recipient Name.
12021	Recipient Email is required.	Recipient Email is empty. Provide a non-empty recipient email.
12022	Recipient Email Length	Recipient Email length exceeds maximum number of characters allowed. Provide a Recipient Email with a valid length.
12023	Invalid Recipient Email.	Recipient Email is not valid. Check Recipient Email follows email address rules.
12024	Email Delivery Failed.	Email Delivery to Payer failed. Check payer email address is valid and active.

#### UnionPay (China)

00 Payment is successful.  
 01 Refer to card issuer.  
 03 Invalid merchant.  
 05 Merchant or seller account closed.  
 06 Issuer card is expired.  
 09 Request in progress.  
 12 Invalid transaction.  
 13 Invalid amount.  
 14 Invalid buyer account or card information.  
 17 Buyer cancel transaction.  
 22 Suspected malfunction.  
 25 Failure to search an original transaction.  
 30 Format error or MAC error.  
 31 Invalid bank.  
 39 No credit account.  
 51 Insufficient fund.  
 53 No saving account.  
 57 Transaction not permitted.  
 61 Withdrawal limit exceeded.  
 65 Withdrawal frequency exceeded.  
 76 Transaction not found.  
 77 Invalid buyer name or buyer Id.  
 79 Host decline when down.  
 85 Internal error at bank's system.  
 87 Debit failed (exception handling).  
 88 Credit failed (exception handling).  
 89 Transaction not received (exception handling).  
 92 Financial institution or intermediate network facility cannot be found for routing.  
 93 Transaction cannot be completed.  
 FE Internal error.  
 OE Transaction rejected as not in operating hours.  
 OF Transaction timeout.

OA Session timeout.  
 1A Buyer session timeout at bank login page.  
 1B Buyer failed to provide the necessary info to login to bank login page.  
 1C Buyer choose cancel at login page.  
 1D Buyer session timeout at account selection page.  
 1E Buyer failed to provide the necessary info at account selection page.  
 1F Buyer choose cancel at account selection page.  
 1G Buyer session timeout at TAC request page.  
 1H Buyer failed to provide the necessary info at TAC request page.  
 1I Buyer choose cancel at TAC request page.  
 1J Buyer session timeout at confirmation page.  
 1K Buyer failed to provide the necessary info at confirmation page.  
 1L Buyer choose cancel at confirmation page.  
 2A Transaction amount is lower than minimum limit.

**Paysbuy (Thailand)**

Error Code	Description	Action / Remark
90	No data is sent	inv is null
91	secureCode is incorrect	secureCode is incorrect
92	PSBID, email address, or password is incorrect	PSBID or email not found
93	PSBID or email address format is incorrect, or password is not entered	PSBID or Username is incorrect

# RESOURCES

## Logos of all brand name

Merchant may download the logos from <https://merchant.razer.com/v3/media-library/>  
password: RazerMerchantServices

## Mobile SDK/XDK

RMS is now ready to be integrated into your mobile apps. We have released Mobile SDK and XDK library on GitHub. To apply this, kindly contact support-sa@razer.com and provide your Merchant ID or Company name, Platform (iOS/Android), Apps Name in order to register and authorize 3<sup>rd</sup> party app in PG system.

## Supported Shopping Cart

RMS have been integrated with many popular shopping carts, globally. Merchant may refer the complete list on <http://www.molpay.com/cart>  
Some payment plugin/add-on/modules can be downloaded from GitHub.

## ISO References

[http://www.iso.org/iso/country\\_codes.htm](http://www.iso.org/iso/country_codes.htm)  
[http://en.wikipedia.org/wiki/ISO\\_3166-1](http://en.wikipedia.org/wiki/ISO_3166-1)  
[http://www.iso.org/iso/currency\\_codes](http://www.iso.org/iso/currency_codes)  
[http://en.wikipedia.org/wiki/ISO\\_4217](http://en.wikipedia.org/wiki/ISO_4217)

## Handling JSON/PLAIN TEXT using .NET

<http://stackoverflow.com/questions/36216464/wcf-webinvoke-which-can-accept-content-type-text-plain>

Please use the above custom WebContentTypeMapper if you are using .NET, especially when you encounter this exception message:

The incoming message has an unexpected message format 'Raw'. The expected message formats for the operation are 'XML'; 'JSON'. This can be because a WebContentTypeMapper has not been configured on the binding. See the documentation of WebContentTypeMapper for more details.