



## **kiple Online Payment Gateway**

# **API Specifications for Merchant v6.13**

(New: Settlement Notification & API)

### **TECHNICAL CONTACT**

Email: [it@webonline.com.my](mailto:it@webonline.com.my)

## *Table of Contents*

TECHNICAL CONTACT .....	1
INTRODUCTION.....	3
INTENDED AUDIENCE AND REQUIRED SKILLS .....	3
KIPLE MERCHANT API OVERVIEW .....	4
KIPLE PAYMENT CONNECTOR (MAIN).....	8
KIPLE PAYMENT CONNECTOR (RETURN PARAMETERS).....	17
KIPLE PAYMENT CALLBACK.....	19
KIPLE PAYMENT SETTLEMENT NOTIFICATION .....	22
KIPLE PAYMENT SETTLEMENT ENQUIRY API .....	24
KIPLE PAYMENT CONNECTOR (QUERY PARAMETERS).....	29
KIPLE PAYMENT PROCESSING .....	32
MERCHANT INTEGRATION TESTING - UAT .....	33
MERCHANT PANEL .....	34
OTHER MATTERS .....	35

## **INTRODUCTION**

kiple Online Payment Gateway is a convenient payment system for merchants. kiple Online Payment Gateway provides Merchants who require credit card or internet banking payment option for their online store on a single platform.

### **Benefits**

- Single API
- Single Platform
- No Maintenance/Monthly Fee
- Low Transaction Fee
- Transaction Account Summary
- Fast Settlement

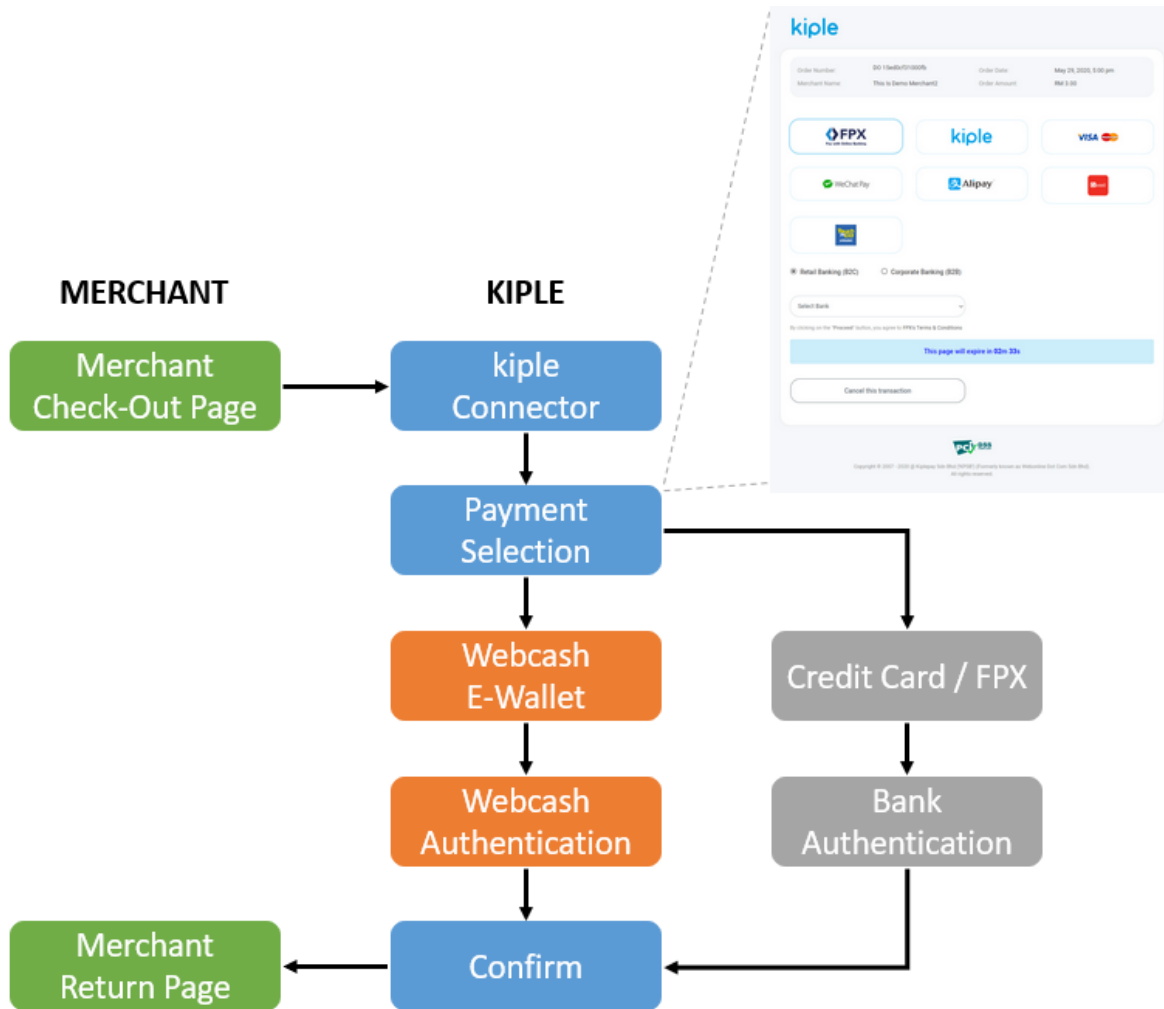
## **INTENDED AUDIENCE AND REQUIRED SKILLS**

This document is intended for software developers with knowledge and authority to integrate third party payment solutions on a Merchant's e-commerce platform. Since kiple Online Payment Gateway does not require any software installation the required skills is determined by the technologies used on the Merchant's website and the necessity to integrate the Payment Gateway with back end systems.

The integration details in this document apply to all Merchants regardless of programming language.

# KIPLE MERCHANT API OVERVIEW

This document is to be used by Merchants of kiple with a valid online account. A typical transaction flow is illustrated in the diagram below:









For the scope of a typical transaction, the scope begins from the confirmed checkout on Merchant side. Merchant website is in a state where all relevant information is captured and ready to present the payment transaction.


## PAYMENT SELECTION PAGE

**kiple**

Order Number: DO15ed0cf31000fb      Order Date: May 20, 2020, 5:00 pm  
Merchant Name: This Is Demo Merchant2      Order Amount: RM 3.00




Retail Banking (B2C)     Corporate Banking (B2B)

Select Bank

By clicking on the "Proceed" button, you agree to Kiple's Terms & Conditions.




**This page will expire in 02m 33s**







Copyright © 2007 - 2020 © Kiplepay Sdn Bhd (KPSB) (Formerly known as WebOnline Dot Com Sdn Bhd). All rights reserved.

**kiple**

Order Number: DO15ed0cf31000fb      Order Date: May 20, 2020, 5:00 pm  
Merchant Name: This Is Demo Merchant2      Order Amount: RM 3.00



Name on Card

Card Number


Expiry Date MM / YY

CVC / CVV

Save this card for faster checkout

By clicking on the "Proceed" button, you agree to Kiple's Terms & Conditions.

**This page will expire in 01m 42s**



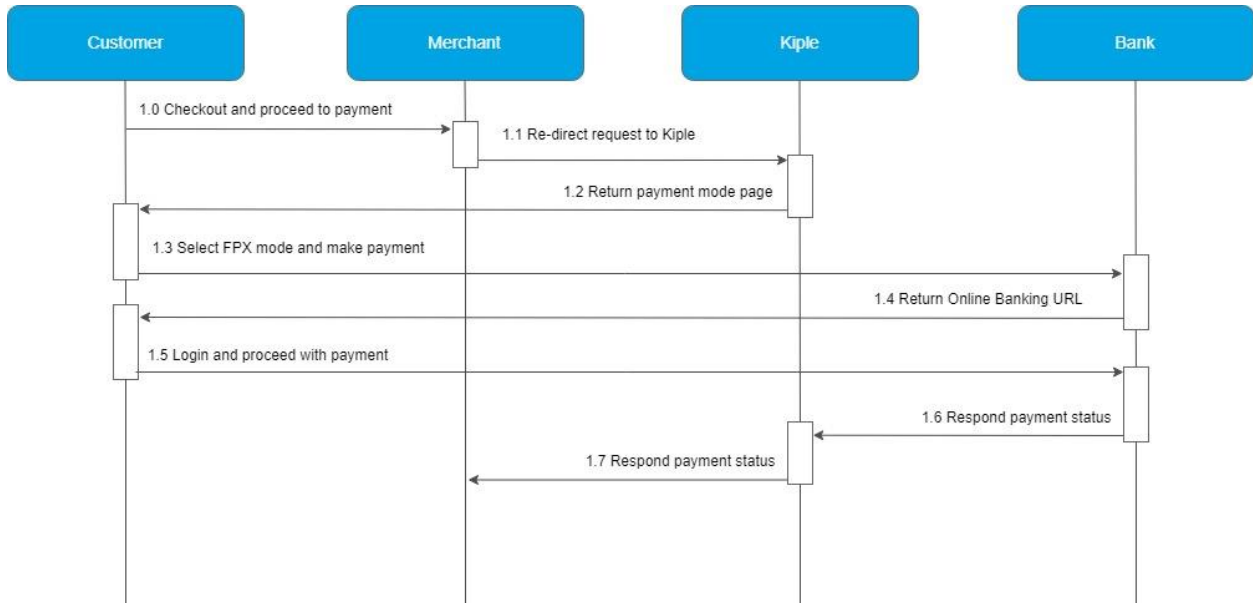
Copyright © 2007 - 2020 © Kiplepay Sdn Bhd (KPSB) (Formerly known as WebOnline Dot Com Sdn Bhd). All rights reserved.

**A payment transaction on kiple Payment Gateway works as follows:**

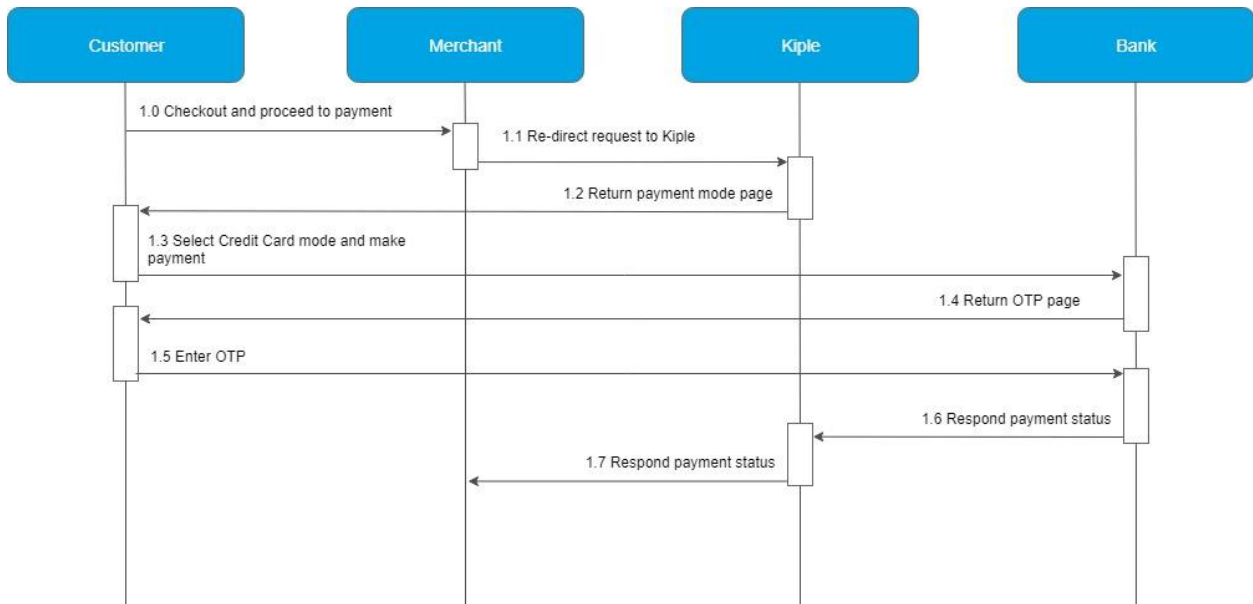
1. Merchant compile all required and relevant optional parameters such that it is to be posted to **kiple Payment Connector**.
2. When Merchant checkout is “linked” to **kiple Payment Connector** (gateway), it will first validate if it is a valid merchant. The connector will show error message if it is a non-Merchant or Merchant which has been suspended.
3. Upon successful validation, it will bring the buyer to the payment channel selection page.
  - a. INTERNET BANKING SERVICES:
    - User can opt for Internet Banking Service, where payment will be deducted from their current / saving account with selected internet banking services.
    - Available Internet Banking Services : CIMBClicks, Maybank2u, Maybank2e, RHB Now, Hong Leong Bank, PBE Bank, Affinbank and Bank Islam.
  - b. KIPLEPAY E-WALLET:
    - Member will have to login with a valid login ID (mobile number or email) and, password as shown on the screen. Upon successful login by kiplePay member, the Payment connector will check if the balance amount is sufficient to satisfy the transaction presented.
    - If the balance is insufficient, member can opt to top-up there and then by selecting their preferred top-up channel (FPX, CIMBCLICKS or MEPSCASH)
4. Upon completion of the payment selection page, **kiple Payment Connector** will handover the control of the transaction back to the Merchant’s specified URL along with the return parameters\*.
5. Merchant Return Page will validate buyer’s order based on Return Code\* received from kiple.
6. kiplePay merchant’s site is allowed to requery kiplePay system to verify completed online transaction information.
7. Buyer could also decide to abort the transaction of which appropriate status will be returned to the Merchant concerned.

\*Please refer to parameter section for details

## PAYMENT FLOW (FPX)



## PAYMENT FLOW (CREDIT CARD)



# KIPLE PAYMENT CONNECTOR (MAIN)

Merchant Checkout Page → kiple Payment Gateway Connector

Note: You must use “POST” method for ‘kiple Payment Connector’

Required Parameters Linking to **kiple Payment Connector**:

Description	Field Name	Type
Order Date	ord_date	Mandatory
Order Total	ord_totalamt	Mandatory. (Must be equal or greater than 1.00)
GST Amount	ord_gstamt	Mandatory (must be 0.00)
Ship to Name	ord_shipname	Mandatory
Ship to Country	ord_shipcountry	Optional
Order Ref	ord_mercref	Mandatory
Telephone	ord_telephone	Optional
E-Mail	ord_email	Optional
Delivery Charges	ord_delcharges	Optional
Service Charges	ord_svccharges	Optional
Custom Field 1	ord_customfield1	Optional
Custom Field 2	ord_customfield2	Optional
Custom Field 3	ord_customfield3	Optional
Custom Field 4	ord_customfield4	Optional
Merchant ID	ord_merclD	Mandatory
Return URL	ord_returnURL	Mandatory
Merchant Hash Value	merchant_hashvalue	Mandatory – Refer notes below
Payment Code	payment_code	Optional – Refer to Payment Code Section
Dynamic callback URL, different from registered URL	dynamic_callback_url	Optional
Version Refer to “Version” section below	version	Optional



### Algorithm to generate merchant\_hashvalue

merchant\_hashvalue = Concatenate the values of merchant\_secret, ord\_mercID, ord\_merceref and ord\_totalamt.

Remove the dot symbol (.) from the ord\_totalamt parameter, if the value contains decimal value

Example : 10.55 => 1055, 25.00 => 2500, 10 => 10.

Sample code in PHP:

```
$amount = str_replace('.', '', $ord_totalamt);  
echo sha1($merchant_secret . $ord_mercID . $ord_merceref . $amount);
```

merchant\_secret = provided by kiple

ord\_mercID = provided by kiple

## Version

Additional data on response will be returned for version greater than or equal to 2.0 onwards.

Version	Response	Description
2.0	payment_type	Payment type of which user used to pay. Available options are: <b>bank</b> : Bank <b>wallet</b> : Kiple Wallet <b>cc</b> : Credit Card <b>wechat</b> : WeChat <b>alipay</b> : Alipay
2.0	source_type	This applies to specific card type, others will return "" (null). <b>LCC</b> : Local Credit Card <b>LDC</b> : Local debit card <b>FC</b> : Foreign Card
2.0	bank	This applies to FPX payment; others will return "" (null). Bank code: e.g: <b>ABB0233</b>
2.0	raw_error_description	Raw description provided by payment connector Kiple connected with.
2.0	bank_reference	This refers to FPX transaction id, only for FPX payment e.g: 2006181725560886
2.0	entry_id	This refers to FPX seller order number, only for FPX payment e.g: FPX13967489

## FPX Bank ID

Below is the list of Bank IDs that can be used in both Staging and Production environment for both B2C (Retail Internet Banking) and B2B (Corporate Internet Banking).

The FPX Bank IDs is output by Kiplepay for v2.0 response parameter: **bank**

### Staging B2C (Individual/Retail Internet Banking)

Bank ID	Bank Name
ABB0233	Affin Bank Berhad
ABMB0212	Alliance Bank Malaysia Berhad
AMBB0209	AmBank Malaysia Berhad
BCBB0235	CIMB Bank Berhad
BIMB0340	Bank Islam Malaysia Berhad
BKRM0602	Bank Kerjasama Rakyat Malaysia Berhad
BMMB0341	Bank Muamalat Malaysia Berhad
BSN0601	Bank Simpanan Nasional
CIT0219	CITI Bank Berhad
HLB0224	Hong Leong Bank Berhad
HSBC0223	HSBC Bank Malaysia Berhad
KFH0346	Kuwait Finance House (Malaysia) Berhad
MB2U0227	Malayan Banking Berhad (M2U)
OCBC0229	OCBC Bank Malaysia Berhad
PBB0233	Public Bank Berhad
RHB0218	RHB Bank Berhad
SCB0216	Standard Chartered Bank
UOB0226	United Overseas Bank
ABB0234	Affin Bank Berhad B2C - Test ID
TEST0021	SBI Bank A
TEST0022	SBI Bank B
TEST0023	SBI Bank C
UOB0229	United Overseas Bank - B2C Test

**Production B2C (Individual/Retail Internet Banking)**

<b>Bank ID</b>	<b>Bank Name</b>
ABB0233	Affin Bank Berhad
ABMB0212	Alliance Bank Malaysia Berhad
AMBB0209	AmBank Malaysia Berhad
BCBB0235	CIMB Bank Berhad
BIMB0340	Bank Islam Malaysia Berhad
BKRM0602	Bank Kerjasama Rakyat Malaysia Berhad
BMMB0341	Bank Muamalat Malaysia Berhad
BSN0601	Bank Simpanan Nasional
CIT0219	CITI Bank Bhd
HLB0224	Hong Leong Bank Berhad
HSBC0223	HSBC Bank Malaysia Berhad
KFH0346	Kuwait Finance House (Malaysia) Berhad
MB2U0227	Malayan Banking Berhad (M2E)
MBB0228	Malayan Banking Berhad (M2U)
OCBC0229	OCBC Bank Malaysia Berhad
PBB0233	Public Bank Berhad
RHB0218	RHB Bank Berhad
SCB0216	Standard Chartered Bank
UOB0226	United Overseas Bank

**Staging B2B (Corporate Internet Banking)**

<b>Bank ID</b>	<b>Bank Name</b>
ABB0232	Affin Bank Berhad
ABB0235	Affin Bank Berhad B2B
ABMB0213	Alliance Bank Malaysia Berhad
AMBB0208	AmBank Malaysia Berhad
BCBB0235	CIMB Bank Berhad
BIMB0340	Bank Islam Malaysia Berhad
BKRM0602	Bank Kerjasama Rakyat Malaysia Berhad
BMMB0342	Bank Muamalat Malaysia Berhad
BNP003	BNP Paribas Malaysia Berhad
CIT0218	CITI Bank Berhad
DBB0199	Deutsche Bank Berhad
HLB0224	Hong Leong Bank Berhad
HSBC0223	HSBC Bank Malaysia Berhad
KFH0346	Kuwait Finance House (Malaysia) Berhad
MBB0228	Malayan Banking Berhad (M2E)
OCBC0229	OCBC Bank Malaysia Berhad
PBB0233	Public Bank Berhad
PBB0234	Public Bank Enterprise
RHB0218	RHB Bank Berhad
SCB0215	Standard Chartered Bank
TEST0021	SBI Bank A
TEST0022	SBI Bank B
TEST0023	SBI Bank C
UOB0227	United Overseas Bank
UOB0228	United Overseas Bank B2B Regional

**Production B2B (Corporate Internet Banking)**

<b>Bank ID</b>	<b>Bank Name</b>
ABB0232	Affin Bank Berhad
ABB0235	Affin Bank Berhad
ABMB0213	Alliance Bank Malaysia Berhad
AMBB0208	AmBank Malaysia Berhad
BCBB0235	CIMB Bank Berhad
BIMB0340	Bank Islam Malaysia Berhad
BKRM0602	Bank Kerjasama Rakyat Malaysia Berhad
BMMB0342	Bank Muamalat Malaysia Berhad
BNP003	BNP Paribas Malaysia Berhad
CIT0218	CITI Bank Bhd
DBB0199	Deutsche Bank Berhad
HLB0224	Hong Leong Bank Berhad
HSBC0223	HSBC Bank Malaysia Berhad
KFH0346	Kuwait Finance House (Malaysia) Berhad
MBB0228	Malayan Banking Berhad (M2E)
OCBC0229	OCBC Bank Malaysia Berhad
PBB0233	Public Bank Berhad
PBB0234	Public Bank Enterprise
RHB0218	RHB Bank Berhad
SCB0215	Standard Chartered Bank
UOB0227	United Overseas Bank
UOB0228	United Overseas Bank B2B Regional

## Payment Code

Below are the list of payment codes available to be processed by Kiplepay.

You need to specify the code below in payment\_code parameter when submitting the form to Kiplepay Payment Gateway.

Payment Code	Payment Processor
CC	Credit Card
AFBB	AFFIN Bank
ALBB	Alliance Bank
AMBB	AMBank
BIBB	Bank Islam
BKRB	Bank Kerjasama Rakyat
BMMB	Bank Muamalat
BSNB	Bank Simpanan Nasional
CIMB	CIMB Clicks
HLBB	Hong Leong
HSBC	HSBC
KFHB	Kuwait Finance House
MYBB	Maybank2u
MYBB2E	Maybank2e
OCBC	OCBC Bank
PUBB	Public Bank
RHBB	RHB Now
SCBB	Standard Chartered
UOBB	United Overseas Bank
DSBB	DEUTSCHE BANK
WECHAT	WECHAT
ALIPAY	ALIPAY
BOOST	BOOST
TNG	TouchNGo
WEBCASH	Kiple

**TYPE Definition:**

Type	Description
Mandatory	Merchant <b>MUST</b> give value for these parameters.
Optional	These are optional parameters.

All the fields above must be presented to **kiple Payment Connector** via HTTP POST method. The fields can be either a normal INPUT field element or HIDDEN INPUT element.

The “Optional” items can be left blank but the “Mandatory” parameters must be presented with valid values.

**Order Ref** is used to track the transaction and therefore each transaction ref. number presented must be unique.

**Return URL** is important so that **kiple Payment Connector** can link to your result processing page.

**Submission URL:** <https://kiplepay.com/wcgatewayinit.php> (LIVE system)

---

REFER TO SECTION **MERCHANT INTEGRATION TESTING – UAT**  
FOR STAGING / TESTING SERVER URL

---



## KIPLE PAYMENT CONNECTOR (RETURN PARAMETERS)

kiple Payment Gateway → Merchant Return Page

Descriptions	Field Name	Type
Order Date	ord_date	Mandatory
Order Total	ord_totalamt	Mandatory
GST Amount	ord_gstamt	Mandatory (must be 0.00)
Ship to Name	ord_shipname	Mandatory
Ship to Country	ord_shipcountry	Optional
Order Ref	ord_mercref	Mandatory
Telephone	ord_telephone	Optional
E-Mail	ord_email	Optional
Delivery Charges	ord_delcharges	Optional
Service Charges	ord_svccharges	Optional
Custom Field 1	ord_customfield1	Optional
Custom Field 2	ord_customfield2	Optional
Custom Field 3	ord_customfield3	Optional
Custom Field 4	ord_customfield4	Optional
Merchant ID	ord_merclD	Mandatory
kiplePay ID	wclD	Returned
*MERCHANT RETURN KEY	ord_key	Returned
RETURN CODE	returncode	Returned
Payment Code	payment_code	Optional
Version (Refer to Versioning table above)	version	Returned

### Algorithm to generate ord\_key

ord\_key = concatenate the values of merchant\_secret, ord\_mercID, ord\_mercID, ord\_mercID, ord\_totalamt and returncode

Remove the dot symbol (.) from the ord\_totalamt parameter, if the value contains decimal value

Example : 10.55 => 1055, 25.00 => 2500, 10 => 10.

Sample code in PHP:

```
$amount = str_replace('.', '', $ord_totalamt);  
echo sha1($merchant_secret . $ord_mercID . $ord_mercID . $ord_mercID . $amount . $returncode);
```

merchant\_secret = provided by kiple

ord\_mercID = provided by kiple

returncode = fixed value

ord\_key will be provided even if the transaction is aborted (E2) or failed (E1).

### TYPE Definition

Type	Description
Mandatory	Merchant <b>MUST</b> give value for these parameters.
Optional	These are optional parameters.
Returned	These parameters are generated by kiple system.

The Return Code takes on the following format:

Return Code	Description
100	Success
E1	Fail
E2	Abort
-1	Validation Error – could be due to missing parameters or invalid parameters given.

## KIPLE PAYMENT CALLBACK

Kiple Payment Callback allows merchant to receive server to server notification via registered merchant URL. Merchant need to provide Kiple with the URL where Kiple will trigger the callback with the information below. This is very useful for cases where merchant didn't receive redirect request from user due to various reasons such as connection timeout, user closed the browser or issue between merchant url and Kiple.

The server to server notification will be in **POST** data format. The notification will be triggered for Success, Fail and Abort statuses. The callback timing slightly varied between each notification posted.

kiple Payment Gateway → Merchant Callback URL

Descriptions	Field Name	Type
Order Date	ord_date	Mandatory
Order Total	ord_totalamt	Mandatory
GST Amount	ord_gstamt	Mandatory (must be 0.00)
Ship to Name	ord_shipname	Mandatory
Ship to Country	ord_shipcountry	Optional
Order Ref	ord_mercref	Mandatory
Telephone	ord_telephone	Optional
E-Mail	ord_email	Optional
Delivery Charges	ord_delcharges	Optional
Service Charges	ord_svccharges	Optional
Custom Field 1	ord_customfield1	Optional
Custom Field 2	ord_customfield2	Optional
Custom Field 3	ord_customfield3	Optional
Custom Field 4	ord_customfield4	Optional
Merchant ID	ord_merclD	Mandatory
kiplePay ID	wclD	Returned
*MERCHANT RETURN KEY	ord_key	Returned
RETURN CODE	returncode	Returned
Dynamic callback URL, different from registered URL	dynamic_callback_url	Optional
Version (Refer to Versioning table above)	version	Returned

### Algorithm to generate ord\_key

ord\_key = concatenate the values of merchant\_secret, ord\_mercID, ord\_mercID, ord\_mercID, ord\_totalamt and returncode

Remove the dot symbol (.) from the ord\_totalamt parameter, if the value contains decimal value

Example : 10.55 => 1055, 25.00 => 2500, 10 => 10.

Sample code in PHP:

```
$amount = str_replace('.', '', $ord_totalamt);  
echo sha1($merchant_secret . $ord_mercID . $ord_mercID . $ord_mercID . $amount . $returncode);
```

merchant\_secret = provided by kiple

ord\_mercID = provided by kiple

returncode = fixed value

ord\_key will be provided even if the transaction is aborted (E2) or failed (E1).

### TYPE Definition

Type	Description
Mandatory	Merchant <b>MUST</b> give value for these parameters.
Optional	These are optional parameters.
Returned	These parameters are generated by kiple system.

The Return Code takes on the following format:

Return Code	Description
100	Success
E1	Fail
E2	Abort

**CALLBACK INTERVALS:**

Kiple will attempt to do server to server notification for a maximum of 6 times. In the case where kiple received response code 200 (HTTP Status Code), notification will be terminated and no further callback will be triggered for the merchant reference.

<b>Tries</b>	<b>Interval</b>
1	1 minute
2	2 minutes
3	4 minutes
4	9 minutes
5	16 minutes
6	25 minutes

# KIPLE PAYMENT SETTLEMENT NOTIFICATION

The settlement notification program notifies the merchant configured URL after settlement by Kiple Finance. The program notifies the merchant every 30 minutes, and if the notification fails, it notifies again 30 minutes later.

Descriptions	Field Name	Type
Event Code	event_code	Mandatory (SETTLEMENT)
Settlement Date	settlement_date	Mandatory
MDR type	type	Mandatory (NORMAL,MULTI-TIER,WO-TPA, W-TPA)
Total Settled Amount	total_settled_amount	Mandatory
Total Settled Transactions	total_settled_transactions	Mandatory
Settlement Details list	settlement_detail	Mandatory
Settlement Type	settlement_type	Mandatory
Settlement Amount	settlement_amount	Mandatory
Settlement Transactions	settlement_transactions	Mandatory
Settlement Summary Id	settlement_summary_id	Mandatory
Signature	_signature	Mandatory (which is alphabetical sorted by the parameters name of data.header)
RETURN CODE	returncode	Returned

### Example:

```
{
  "event_code": "SETTLEMENT",
  "data": {
    "header": {
      "settlement_date": "2020-07-20",
      "type": "NORMAL",
      "total_settled_amount": "18092.00",
      "total_settled_transactions": "20",
      "settlement_detail": [
        {
          "settlement_type": "lcc",
          "settlement_amount": "16090.00",
          "settlement_transactions": "17",
          "settlement_summary_id": "33"
        },
        {
          "settlement_type": "ldc",
          "settlement_amount": "2001.00",
          "settlement_transactions": "2",
          "settlement_summary_id": "34"
        },
        {
          "settlement_type": "fc",
          "settlement_amount": "1.00",
          "settlement_transactions": "1",
          "settlement_summary_id": "35"
        }
      ]
    }
  },
  "_signature": "eac5e767f65926452d2e08d30c52e6329281b621"
}
```

### Remark:

```
_signature = sha1(merchant_id + merchant_hash_key + event_code + hash_string)
event_code = SETTLEMENT
hash_string = 2020-07-20
[{"settlement_type": "lcc", "settlement_amount": "16090.00", "settlement_transactions": 17, "settlement_summary_id": 33}, {"settlement_type": "ldc", "settlement_amount": "2001.00", "settlement_transactions": 2, "settlement_summary_id": 34}, {"settlement_type": "fc", "settlement_amount": "1.00", "settlement_transactions": 1, "settlement_summary_id": 35}]18092.0020NORMAL
```

(From data.header parameters value =  
settlement\_date+settlement\_detail+total\_settled\_amount+total\_settled\_transactions+type)

## KIPLE PAYMENT SETTLEMENT ENQUIRY API

POST /api/v2/thirdparty/settlement-enquiry.json

Obtain list of settlement details performed by the merchant call. This API will list 100 records paginated results, for subsequent records, you can specify page parameters in to change the page.

### Example URL:

[https://webcash.com.my/api/v2/thirdparty/settlement-enquiry.json?  
merchant\\_id=80007536&hash\\_key=e4d8a35454eac0bab5ce9e12d172349d05e5cc02&  
page=1&limit=100&type=NORMAL](https://webcash.com.my/api/v2/thirdparty/settlement-enquiry.json?merchant_id=80007536&hash_key=e4d8a35454eac0bab5ce9e12d172349d05e5cc02&page=1&limit=100&type=NORMAL)(live system)

---

REFER TO SECTION **MERCHANT INTEGRATION**  
<uat.kiplepay.com/staging-cn.webcash.com.my>  
FOR STAGING / TESTING SERVER URL

---



### Query String (URL) Request

Parameters	Required	Description
page	O	page number Values: 1, 2, 3...
limit	O	limit per page, by default is 100 and maximum is 200.
type	O	MDR type (Case insensitive) Values: NORMAL, MULTI-TIER, WO-TPA, W-TPA Default value: NORMAL
merchant_id	M	Master merchant ID
hash_key	M	Request security verification key  Rules: Use SHA1 to encrypt the following strings: merchant ID + Merchant Hash Key + "settlement-enquiry" + hash String (Hash String is the concatenated string from Form Data (POST) Request which is alphabetical sorted by the parameters name )  For example: Merchant id :80009974 Merchant Hash Key :3R8wHekUoxET74a5 Form Data (POST) Request: settlement_date:2020-07-20 settlement_summary_id:34,32,31,30,33 start_date:2020-05-01 end_date:2020-07-19  Hash String will be "end_date +settlement_date + settlement_summary_id + start_date"  Hash_key = SHA1(800099743R8wHekUoxET74a5settlement- enquiry2020-07-192020-07-2034,32,31,30,332020-05-01) = b4359628a833ce127b1fb42c781790bb365cec95

### Form Data (POST) Request

Parameters	Required	Description
start_date	O	Start date of search – format: YYYY-MM-DD
end_date	O	End date of search – format: YYYY-MM-DD
settlement_summary_date	O	Settlement date of search – format: YYYY-MM-DD
settlement_summary_id	M	Settlement Summary ID (Batch ID) (xx or xx,xx,xx <a href="#">MULTI-TIER only support single</a> )

Error Code	Description
invalid_access	Invalid Merchant ID
invalid_key	Invalid Hash Key
invalid_settlement_summary_id	Invalid Settlement Summary Id
invalid_type	Invalid MDR Type
404	Page Not Found

### RESPONSE

Blank fields are included as null instead of being omitted. All timestamps are returned in ISO 8601 format: YYYY-MM-DDTHH:MM:SSZ

Output are categorized according to the four parameters below:

Response Key parameter calculation

Partner ID + Merchant Secret Key + Action Endpoint + POST parameters data sorted by alphabetical + 100

Example:

POST parameters data:2020-07-192020-07-2034,32,31,30,332020-05-01

Final hash string will be

800099743R8wHekUoxET74a5settlement-enquiry2020-07-192020-07-2034,32,31,30,332020-05-01100

response\_key will be SHA1(hash string) = 91fbc12012b5da1545342f06c85c9b19f584d8b5

### API Response Example

POST: /api/v2/thirdparty/settlement-

enquiry.json?merchant\_id=80009974&hash\_key=20c2bf71d130d3670d7135288476b43aeb327031&page=5&limit=2&type=NORMAL

## Success:

```
{
  "success": true,
  "data": {
    "header": {
      "current_page": 5,
      "total_pages": 9,
      "current_records": 2,
      "total_records": 18,
      "total_amount": "16118.00"
    },
    "results": [
      {
        "webcash_reference": 24258965,
        "merchant_reference": "click-bb49AFFYos10pLa5-3crcL0vJa97Rfya5",
        "bank_reference": "0",
        "webcash_id": "80006492",
        "settled_date": "2020-07-20T00:00:00+08:00",
        "settled_type": "lcc",
        "date": "2020-05-22T18:31:27+08:00",
        "merchant_id": "80007535",
        "merchant_name": "Nadira Corner",
        "buyer_name": "Frankie",
        "buyer_email": "fx.zhu@greenpacket.com.cn",
        "buyer_telephone": "60003",
        "gross_amount": "3000.00",
        "mdr_amount": "2.00",
        "refund_amount": "0.00",
        "fee_gst": "0.00",
        "net_amount": "2998.00",
        "custom_field_1": "1ClickPay",
        "custom_field_2": null,
        "custom_field_3": "3",
        "custom_field_4": null
      },
      {
        "webcash_reference": 24258967,
        "merchant_reference": "click-bb49AFFYos10pLa5-k9S66Z9TTwvALta5",
        "bank_reference": "0",
        "webcash_id": "80006492",
        "settled_date": "2020-07-20T00:00:00+08:00",
        "settled_type": "lcc",
        "date": "2020-05-22T18:43:16+08:00",
        "merchant_id": "80007535",
        "merchant_name": "Nadira Corner",
        "buyer_name": "Frankie",
        "buyer_email": "fx.zhu@greenpacket.com.cn",
        "buyer_telephone": "66",
        "gross_amount": "2000.00",
        "mdr_amount": "2.00",
        "refund_amount": "0.00",
        "fee_gst": "0.00",
        "net_amount": "1998.00",
        "custom_field_1": "1ClickPay",
        "custom_field_2": null,
        "custom_field_3": "2",
        "custom_field_4": null
      }
    ]
  },
  "code": "success",
  "message": "Success",
  "response_key": "36e24deffd4461836e26d9c5513db91247cdd87a"
}
```

**Error:**

Invalid Merchant ID:

```
{
  "success": false,
  "data": [],
  "code": "invalid_access",
  "message": "Invalid Merchant ID"
}
```

Invalid Hash Key:

```
{
  "success": false,
  "data": [],
  "code": "invalid_key",
  "message": "Invalid Hash Key"
}
```

Invalid MDR Type:

```
{
  "success": false,
  "data": [],
  "code": "invalid_type",
  "message": "Invalid MDR Type"
}
```

Invalid Settlement Summary Id:

```
{
  "success": false,
  "data": [],
  "code": "invalid_settlement_summary_id",
  "message": "Invalid Settlement Summary Id"
}
```

Page large than total Page:

```
{
  "message": "NotFound",
  "url": "/api/v2/thirdparty/settlementEnquiry.json?merchant_id=80009974&hash_key=0dabba53843666e0d98403e0c91ea1e9cadcb912&page=10&limit=2",
  "code": 404
}
```

# KIPLE PAYMENT CONNECTOR (QUERY PARAMETERS)

Merchant Return Page → kiple Payment Gateway

Descriptions	Field Name	Type	Required
Merchant ID	ord_merclD	Returned	Required
Order Ref	ord_mercref	Returned	Required
Order Total	ord_totalamt	Returned	Required
Version	version	Returned	Optional

## TYPE Definition

Type	Description
Returned	These parameters are the same value generated by merchant and returned back to merchant by kiple

## Response on Unversioned or Version < 2.0

Directly **reply** on the page from Kiple:

### Format: Text

Text Reply from Kiple	Description
S	Success (Payment Received)
P	Transaction Incomplete (Pending)
F	Fail (Payment Failed)
A	User Abort Transaction (Fail)
Invalid Parameters	Invalid Parameters
Record Not Found	Record Not Found

**Response on Version 2.0**

**Format: JSON**

<b>Fields</b>	<b>Description</b>												
payment_type	Payment type of which user used to pay. Available options are: <b>bank:</b> Bank <b>wallet:</b> Kiple Wallet <b>cc:</b> Credit Card <b>wechat:</b> WeChat <b>alipay:</b> Alipay												
source_type	This applies to specific card type, others will return ". <b>LCC:</b> Local Credit Card <b>LDC:</b> Local debit card <b>FC:</b> Foreign Card												
bank	This applies to FPX payment; others will return ". Bank code: e.g: <b>ABB0233</b>												
raw_error_description	Raw description provided by payment connector Kiple connected with.												
bank_reference	This refers to FPX transaction id, only for FPX payment e.g: 2006181725560886												
entry_id	This refers to FPX seller order number, only for FPX payment e.g: FPX13967489												
Status	List of statuses <table border="1"><tbody><tr><td>S</td><td>Success (Payment Received)</td></tr><tr><td>P</td><td>Transaction Incomplete (Pending)</td></tr><tr><td>F</td><td>Fail (Payment Failed)</td></tr><tr><td>A</td><td>User Abort Transaction (Fail)</td></tr><tr><td>Invalid Parameters</td><td>Invalid Parameters</td></tr><tr><td>Record Not Found</td><td>Record Not Found</td></tr></tbody></table>	S	Success (Payment Received)	P	Transaction Incomplete (Pending)	F	Fail (Payment Failed)	A	User Abort Transaction (Fail)	Invalid Parameters	Invalid Parameters	Record Not Found	Record Not Found
S	Success (Payment Received)												
P	Transaction Incomplete (Pending)												
F	Fail (Payment Failed)												
A	User Abort Transaction (Fail)												
Invalid Parameters	Invalid Parameters												
Record Not Found	Record Not Found												

**Query URL:** <https://kiplepay.com/enquiry.php> (LIVE system)

**Query URL:** <https://uat.kiplepay.com/enquiry.php> (Staging system)

## SAMPLE HTML SOURCE FOR PAYMENT CONNECTOR

Merchant Checkout Page → kiple Payment Gateway Connector

```
<HTML>
<BODY>
<form action="https://uat.kiplepay.com/wcgatewayinit.php" method="post">
  <input type="hidden" name="ord_date" value="11 November 2010">
  <input type="text" name="ord_totalamt" value="100.00" />
  <input type="text" name="ord_gstamt" value="0.00" />
  <input type="hidden" name="ord_shipname" value="John Doe">
  <input type="hidden" name="ord_shipcountry" value="Malaysia">
  <input type="hidden" name="ord_merceref" value="Order 32212">
  <input type="hidden" name="ord_telephone" value="0172222222">
  <input type="hidden" name="ord_email" value="demo@xyz.com.my">
  <input type="hidden" name="ord_delcharges" value="0.00">
  <input type="hidden" name="ord_svccharges" value="0.00">
  <input type="text" name="ord_mercID" value="80000321">
  <input type="text" name="merchant_hashvalue" value="187d8cf8b17161772524c6288ecd69885c862dc1">
  <input type="hidden" name="ord_returnURL" value="http://yourdomain.com.my/return.php">
  <input type="submit" name="submit" value="Pay with kiplePay">
</form>
</BODY>
</HTML>
```

## SAMPLE OF RETURN PARAMETERS

kiple Payment Gateway → Merchant Return Page

```
Array (
  [ord_date] => 11 November 2010
  [ord_totalamt] => 100.00
  [ord_gstamt] => 0.00
  [ord_shipname] => John Doe
  [ord_shipcountry] => Malaysia
  [ord_merceref] => Order 32212
  [ord_telephone] => 0172222222
  [ord_email] => demo@xyz.com.my
  [ord_delcharges] => 0.00
  [ord_svccharges] => 0.00
  [ord_key] => 5a3af903adf343cd6d46e2e7c2a98f98632ce78c
  [ord_mercID] => 80000155
  [wcID] => 109384
  [returncode] => 100
)
```

## **KIPLE PAYMENT PROCESSING**

kiple is operating on a different approach to Credit Card type bill presentment. kiple is based on the principle that the Account Holder is empowered to safeguard their Credit Balance such that authorization has to be granted by the Account Holder before transaction can occur.

This is different from Credit Card type transaction where the Card Number is presented for payment and Credit Card company authorizes the transaction.

Such Authorization is done through Account Holder entering the password on all payment transactions. Exception is valid only when the Account Holder has authorized a standing instruction for some type of transaction to be deducted.

As such, the Account Holder is advised to always maintain absolute secrecy of its Account Password as that is the one that authorizes the transaction.



# MERCHANT INTEGRATION TESTING - UAT

Merchants can check if their website payment integration with kiple is working by using kiple's test system. The following are the details for testing.

In UAT mode, you can simply choose any of the banks to complete the payment without going to actual bank site. kiplePay uses simulator to simulate the success test result.

**Submission URL:** <https://uat.kiplepay.com/wcgatewayinit.php>

## Merchant Test Account

Merchant ID	Merchant Secret
80000155	123456

## kiple Member Test Account

Account ID	Password
20000030	tmp1234

## Credit Card Test Account

Test Cards	Card Number	3D Secure Enrolled
Mastercard	5123450000000008	Y
	2223000000000007	Y
Visa	4508750015741019	Y

Expiry Date	Transaction Response Gateway Code
05 / 21	APPROVED
05 / 22	DECLINED

CSC/CVV	CSC/CVV Response Gateway Code
100	MATCH
101	NOT_PROCESSED
102	NO_MATCH

# MERCHANT PANEL

Using login URL below, merchant view how does their panel looks like:

**Login URL:** <https://uat.kiplepay.com/users/login>

## Merchant Test Account

Username	Password
80000155	tmp1234

## kiple Member Test Account

Username	Password
20000030	tmp1234

## OTHER MATTERS

Merchants can link to **kiple Payment Connector** via HTTPS protocol. Merchants can opt to have the standard connector or customized connector. Customized connector involves Merchants working out the layout of the page and to be passed on to kiple for transaction flow consistency.

Terms and Conditions of using **kiple Payment Connector** is governed by the Merchant's Agreement.

Merchants will view transactions with account holders in 'Transaction History' page.