

Eximbay Online Payment

Technical Integration Guide



Contact Information

Technical Support Email : TechnicalSupport@eximbay.com
Operation Support Email : OnlineSupport@eximbay.com

If you have any suggestion or remark about this document please write an email to us.

Copyright

© 2014 EXIMBAY, Co., Ltd. All rights reserved. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of EXIMBAY.

EXIMBAY may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement form EXIMBAY, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Contents

1 Overview	6
1.1 Communication Method	6
1.1.1 BasicProcessor.krp	6
1.1.2 DirectProcessor.krp	6
1.2 Data Format.....	7
2 Online Payment	8
2.1 Performing a Sale & Authorizing a Payment	10
2.1.1 Request URL	10
2.1.2 Request Parameters	10
2.1.3 Response Parameters	14
2.2 Payer Authentication	16
2.2.1 Request URL	16
2.2.2 Request Parameters	16
2.2.3 Response Parameters	16
2.3 Performing a Sale and Authorizing a Payment with Payer Authentication.....	17
2.3.1 Request URL	17
2.3.2 Request Parameters	17
2.3.3 Response parameters	18
2.4 Capturing an Authorization.....	19
2.4.1 Request URL	19
2.4.2 Request Parameters	19
2.4.3 Response parameters	19
2.5 statusurl	21
2.6 Cash Payment Notice.....	22
3 Common Management Interfaces	23
3.1 Crediting a Payment	23
3.1.1 Request URL	23
3.1.2 Request Parameters	23
3.1.3 Response Parameters	24
3.2 Querying a Single Transaction	26
3.2.1 Request URL	26
3.2.2 Request Parameters	26
3.2.3 Response Parameters	27
3.3 Querying a Cash Receipt Transaction	29
3.3.1 Request URL	29
3.3.2 Request Parameters	29
3.3.3 Response Parameters	29
4. Secondary Partner Preferential Fee Differential Settlement Process.....	31
4.1 Performing a Sale (PG Secondary Partner Additional Parameters).....	31

4.1.1 Request URL	31
4.1.2 Request Parameters	31
4.1.3 Response Parameters	32
4.2 Crediting a Payment (Additional Parameters)	32
4.2.1 Request URL	32
4.2.2 Request Parameters	32
4.2.3 Response Parameters	32
5 References	33
5.1 FGKEY	33
5.1.1 How it works	33
5.1.2 Setting up the SHA-256 hashing.....	33
5.1.3 Hash via the merchant	33
5.2 Response Code	34
5.3 AES 256 Encryption.....	34
5.4 About app-to-app	34
5.4.1 IOS.....	34
5.4.2 ANDROID	35
Appendix A Supported Currencies	38
A.1 Request Currencies (cur)	38
Appendix B Supported Languages.....	38
Appendix C Payment Methods (paymethod)	39
Appendix D State, Province and Territory Codes.....	40
D.1 For the United States and Canada.....	40
Appendix E Bank Code.....	42
Appendix F Cash receipt issuance status	43

Recent revisions

Date	Description
2017.04.11	Create Document (Based on General Merchant v2.3 rev2)
2017.08.17	Appendix E PayPal Add Request Parameters for Airline (PAYMENTREQUEST_0_AIRLINE_TICKETNUMBER)
2017.11.09	IFrame not supported
2018.04.26	3.1 Remove returnurl on request for crediting a payment
2018.09.07	Add new payment method TOSS
2018.11.39	Add PG Subcontractor Preferential Fee Rate Application Process
2019.02.08	Add new payment method BankPay
2019.03.27	Add new payment method Yandex
2019.04.01	Add Parameters DM Airline
2019.07.15	Add PAYER_AUTH, PAYMENT_PA, AUTHORIZE_PA
2019.08.05	Add Token Creation, Token Payment
2019.08.22	Add new payment method KakaoPay
2019.10.24	Rename Table of Contents 2.1, 2.3, Add CAPTURE
2020.01.17	Add new payment method and parameters Yandex, Add Appendix H Bankcode Remove Molpay Payment Method from Cash Payment
2020.05-13	Separation of Global and local integration guide
2020-08-06	Add new parameter callfromapp
2020-08-21	Add new parameter item_#_ink only for open market
2020-08-25	Add new payment method smile pay
2020-09-22	Added new parameters for duty-free shops only. (dfs_deliverer, dfs_delivery, dfs_delivery_date)
2020-10-05	Adding new parameters for airlines (traveldt, ticketnum, passengernm, ticketissueagentid, leaveplacecd, flightnm)
2021-03-12	Adding new parameters for domestic payment (toss, kakaopay, payco) cash receipt (cashReceiptType, amt_taxFree, amt_taxable, amt_vat, amt_serviceFee)
2021-03-29	Add Cash Receipt Query API
2021-11-17	Remove HanaSK Card payment Method
2022-02-23	Updated (up to 2022-01-26 of Eximbay Online Local Payment Integration Guide v2_3_kr – rev 11)
2022-04-14	Change App-to-app android settings

1 Overview

This document explains how to link **online domestic payments**. Once the development is completed according to this document, please reply to the contact information you received and you can use the actual service after changing the URL and related parameters.

1.1 Communication Method

1.1.1 BasicProcessor.krp

When EndPoint is BasicProcessor.krp, the request data for each task is requested by the URL defined by HTTP POST method through web-based communication, and the response data is returned by the user browser-based method(returnurl) and the web server-based method (statusurl) through the back-end .



The browser based Front-end(returnurl) URL

URL where a user will be redirected to upon payment completion and payment result will be returned to merchant's page in the browser. User might close their browser any time throughout the payment process. As a result, merchant might not receive any payment result with the URL. Therefore, it is recommended to use this URL only for displaying the payment result to the customer.



The server based Back-end(statusurl) URL

URL where payment result will be sent from Eximbay server to merchant's server in background. This is a back-end server-to-server operation that javascript, cookie and session could not be used. It is recommended to use this URL to update payment status in merchant system and manipulate DB based on the response parameters through statusurl.

Due to Eximbay notification mechanism, same payment result can be sent with the Back-end URL(statusurl) multiple times to the merchant system. Therefore, it is advised to prevent duplication in updating merchant system.

1.1.2 DirectProcessor.krp

When EndPoint is DirectProcessor.krp, it is web-based communication, and request data for each task is requested through the URL defined by HTTP POST method.

And it returns the response value in the HTTP GET format method of text method as a result value without providing UI.

(data format : name1=value1&name2=value2&.....)

1.2 Data Format

Field Name starts with a lowercase letter and it is case sensitive.

Data Type	Description
Integer	Whole number {..., -3, -2, -1, 0, 1, 2, 3, ...}
String	Sequence of letters, numbers, spaces, and special characters

"R" - Required, "C" - Conditional, " " – Optional.

2 Online Payment

Online payment in this document refers to credit card + online Third-party payments and provides an integrated linked document

Online third-party payments means the payment method provided by Eximbay excluding credit cards such as Paypal, Alipay, Wechat, Tenpay, etc.



Unable to support iframe, please use *displaytype='P'* or *displaytype='R'*.

Credit Card

For **credit card** payments, the authentication, authorization and capture are requested in one transaction at the same time. And the transaction is automatically captured on the next day.

Summary of the online Credit Card Payment process

txntype	UI	Request URL	Summary
PAYER_AUTH	O	/BasicProcessor.krp	Card Issuer Authentication(3D) process (DM/DCC can be optionally processed)
AUTHORIZE_PA	X	/DirectProcessor.krp	Credit Card Authorization process. EXIMBAY does not capture the authorization until CAPTURE request is received or merchant separately do the capture.
AUTH_REVERSAL	X	/DirectProcessor.krp	Authorization Reversal process (Only available before capture) If merchant separately do the capture, it cannot be used.
CAPTURE	X	/DirectProcessor.krp	Authorization capture request. Automatically processed in +1 working day.

As mentioned above, there are 4 types of transactions(txntype), and they can be combined as in the below table.

txntype	UI	Request URL	Combination
AUTHORIZE	O	/BasicProcessor.krp	PAYER_AUTH & AUTHORIZE_PA

PAYMENT_PA	X	/DirectProcessor.krp	AUTHORIZE_PA & CAPTURE
PAYMENT	O	/BasicProcessor.krp	PAYER_AUTH & AUTHORIZE_PA & CAPTURE

Note. 1 –Based on *txntype*, request/response parameters can be different. For more information, you can contact us through the contact details in Contact Information.

Online Third-Party Payments

The authentication, authorization and capture are performed in one transaction from the payment processor.

Summary of the online third-party payment process

txntype	UI	Request URL	Combination
PAYMENT	O	/BasicProcessor.krp	PAYER_AUTH & AUTHORIZE_PA & CAPTURE

Note. 1 – Online third-party payments include toss, kakaopay etc... It means the payment method provided by eximbay excluding credit card.

Note. 2 – For third-party payments, all authentication, authorization and capture are performed by a payment processor.

2.1 Performing a Sale & Authorizing a Payment

Using the UI provided by Eximbay, you can perform two approval tasks.

PAYMENT includes **CAPTURE** work and automatic billing will be carried out next day.

AUTHORIZE is charged only when purchased directly from a merchant or when **CAPTURE** is sent to Eximbay.



In the case of using domestic payments in iPhone and Android-based merchant apps, an additional work is required to call the payment company's app. (Refer to 5.4 in the guide)

2.1.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/BasicProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/BasicProcessor.krp>

2.1.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"PAYMENT" or "AUTHORIZE"
ref	String	30	R	Merchant-generated unique order number or tracking number for each transaction including failed one. (e.g.orderid)
cur	String	3	R	ex. USD, SGD, KRW...(refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
amt_taxFree	String	15	C	Tax-free amount of the payment amount. (Do not use comma ',') Total supply price of duty-free goods
amt_taxable	String	15	C	Taxable amount of the payment amount.. (Do not use comma ',') Total supply price of taxable goods
amt_vat	String	15	C	VAT of the payment amount (Do not use comma ',') Total VAT on taxable goods.
amt_serviceFee	String	15	C	Service fee of the payment amount amount (Do not use comma ',')
paymethod	String	4		Payment Method Code (Refer Appendix C)

				This parameter is used when you want to call specific payment method.
shop	String	255		Shop Name(Required when Shop name is different from Merchant name)
buyer	String	64	R	Buyer Name (Required to use real name)
tel	String	32		Buyer's Contact Number
email	String	32	R	Buyer's email address(Buyer will receive email notification)
lang	String	2	R	Payment screen language (Refer to Appendix B)
returnurl	String	255	R	Merchant page that is called when the user exits the payment screen on the payment result confirmation screen (Since returnurl operates based on the customer's browser, it may not be called if the browser is forcibly closed.)
statusurl	String	255	R	After payment processing is completed, it is the merchant page called from Backend, and the returnurl and parameter are the same. Since it is not called from the browser, scripts, cookies, and sessions cannot be used. DB operation and payment process must be handled by statusurl, and returnurl may not be called depending on the payment method or when the customer forcibly closes the payment screen. Since statusurl can be called in duplicate, please make sure that your order is not processed in duplicate.
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
charset	String	32		Default charset – UTF-8 (Refer to Appendix C)
fgkey	String	64	R	Refer to "5.1 FGKEY"
partnercode	String	10		Partner Code (assigned by EXIMBAY)
item_#_product	String	255	R	Item Name
item_#_quantity	String	10	R	Item Quantity (Positive number not less than 1)
item_#_unitPrice	String	15	R	Item Price (Do not use comma ','. Must be more than 0)
Item_#_link	String	255	R	Item Link (Open market)
surcharge_#_name	String	255		Additional charge item name (e.g. discount(-), shipping fee(+) etc..)
surcharge_#_quantity	String	10		Additional charge item quantity (Must be more than 1)

surcharge_#_unitPrice	String	15		Discount amount per item (Do not use comma ',' and negative amount allowed) (e.g. -1000.50, 9.15)
shipTo_city	String	50	R	Shipping city (e.g. Hanoi, Brisbane, Houston)
shipTo_country	String	2	R	Shipping country. Must be ISO3166 country code (e.g. KR, US..)
shipTo_firstName	String	60	R	First name of the person receiving product
shipTo_lastName	String	60	R	Last name of the person receiving product
shipTo_phoneNumber	String	15	R	Phone number for the shipping address(Including country code)
shipTo_postalCode	String	20	R	Postal code for the shipping address
shipTo_state	String	20	R	State or province of the shipping address. Required only when shipping country is US or CA. (e.g. MA, NY, CA) (Refer to Appendix D)
shipTo_street1	String	100	R	Street of the shipping address (e.g. 123 Main street, 56 Le Loi street)
billTo_city	String	50		Billing city (e.g. Hanoi, Brisbane, Houston)
billTo_country	String	2		Billing country. Must be ISO3166 country code (e.g. KR, US..)
billTo_firstName	String	60		Cardholder first name
billTo_lastName	String	60		Cardholder last name
billTo_phoneNumber	String	15		Cardholder contact number(including country code)
billTo_postalCode	String	20		Postal code for the billing address
billTo_state	String	20		State or province of the billing address. Only used for US or CA. (e.g. MA, NY, CA) (Refer to Appendix D)
billTo_street1	String	100		Street of the billing address(e.g. 123 Main street, 56 Le Loi street)
dfs_deliverer	String	100	R	For duty-free use-shipper (requires duty-free shop)
dfs_delivery	String	400	R	Duty free-delivery (required at duty-free shops)
dfs_delivery_date	String	8	R	For duty-free use-Delivery date (required for duty-free shops) (ex. 20200101)
travelDt	String	6	R	For airline-travel date (airline required) (ex. 200101)
ticketnum	String	14	R	For airline-ticket number (airline required)
passengernm	String	100	R	For airline-passenger name (airline required)
ticketissueagentid	String	50	R	For airlines-Ticket issuing agency ID (required by airline)
leaveplaced	String	3	R	For airlines-Departure (destination) (required by airline)

flightnm	String	6	R	For airline-flight number (airline required)
ostype	String	1		P : pc version((default)) M : mobile
displaytype	String	1	R	P : popup (default) R : page redirect
issuercountry	String	2		"KR"
callfromapp	String	1		Classification of client environment Y : When called in Android and IOS APP environment N : When called from a web browser environment Default : N

Note. 1 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document..

Note. 2 – The *ref* parameter is allowed to have duplicate values. You can contact us If you want to prevent from duplicate payments in accordance with *ref*.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters.

Note. 4 – The *partnercode* is used for specific process required for a particular merchant. And the merchant need to consult with us before using this parameter.

Note. 5 – *item_#* - you can specify this parameter up to item numbers. *surcharge_#* - you can specify this parameter up to additional item-specific discounts or charges. # starts from 0.

(*item_#_quantity* * *item_#_unitPrice*) + (*surcharge_#_quantity* * *surCharge_#_unitPrice*) must be equal to *amt*.

Note. 6 – The shipping address *shipTo_** parameters are required in order to prevent from chargebacks. For more information, you can ask detail information from our Operation Support team.

Note. 7 – Do not **use amt_taxable** if the payment method is kakaopay.

Note. 8 – When using online third-party (toss, kakaopay, payco), the *amt_taxfree*, *amt_taxable*, *amt_vat*, *amt_serviceFee* parameters are not required, and when sending parameters, they are sent to each payment company without processing them.

Note. 9 – When using naverpay point as a payment method, the values **amt_taxfree**, **amt_taxable**, **amt_vat** are required and are passed to the payment company without processing.

Note 10 – When using naverpay, please refer to the below for parameters *item_#_produc*, *item_#_quantity*, *item_#_unitPrice* and make a request so that it won't be a problem.

ex) Requesting payment for 3 products

Propolis 1000mg 200capsules, quantity 3

Kangaroo essence 6000mg 60capsules *3pack, quantity 1

Emu Heat Oil 1000ml, quantity 2

item_0_product= Propolis 1000mg 200capsules

item_0_quantity=3

item_0_unitPrice=11893

item_1_product= Kangaroo essence 6000mg 60capsules *3pack

item_1_quantity=1

item_1_unitPrice=1000

item_2_product= Emu Heat Oil 1000ml

item_2_quantity=2

item_2_unitPrice=2000

ex) Requesting payment for 1 product

Propolis 1000mg 200capsules, quantity 1

item_0_product= Propolis 1000mg 200capsules

item_0_quantity=1

item_0_unitPrice=11893

Note 11 – *buyer* is required for using a virtual account, and cannot be changed after requesting payment.

2.1.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"PAYMENT"
ref	String	30	R	Merchant-generated unique order number or tracking number. Same value returns.
cur	String	3	R	e.g. USD, SGD, KRW...(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
email	String	32	R	Buyer Email Address
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
transid	String	24	R	Transaction ID
rescode	String	6	R	Response Code. (Success:'0000', Refer to Chapter 4)
resmsg	String	700	R	Response Message
authcode	String	8	C	Approval Code (Only provided for credit card payment)
resdt	Integer	14	R	YYYYMMDDHHMMSS
cardholder	String	50	C	Cardholder's Name in English
accesscountry	String	2	R	Customer's accessed country. ISO 3166 country codes (e.g. KR, JP, US...)
fgkey	String	64		Refer to "5.1 FGKEY"
cardno1	Integer	4	C	First 4 digits of card number
cardno4	Integer	4	C	Last 4 digits of card number
paymethod	String	4	C	Payment Method Code (Refer to Appendix C)
paymentType	String	10	C	When using online third-party (toss, kakaopay, payco), it responds according to the merchant setting.
payto	String	64	C	Merchant Name that receiving payment
shipTo_city	String	50		Shipping city (e.g. Hanoi, Brisbane, Houston)
shipTo_country	String	2		Shipping country. ISO3166 country code (e.g. KR, US..)
shipTo_firstName	String	60		First name of the person receiving product
shipTo_lastName	String	60		Last name of the person receiving product
shipTo_phoneNumber	String	15		Phone number for the shipping address(Including country code)
shipTo_postalCode	String	20		Postal code for the shipping address

shipTo_state	String	20		State or province of the shipping address. Available only if shipping country is US or CA. (e.g. MA, NY, CA)
shipTo_street1	String	100		Street of the shipping address (e.g. 123 Main street, 56 Le Loi street)
status	String	20	C	(Cash payment) "Registered" or "Sale" Registered means an order registration, only when "Sale" status is returned, you need to consider that the order is completed. "Sale" means that payment is completed. And this status is returned with statusurl. For other payment statuses, refer to 2.1.6 Cash Payment Notice.
inst	String	2	C	Installment months (If it's a domestic payment, create it. 00 if lump sum)
cashreceipt	String	1		Whether to use cash receipt or not (Y/N)
receiptinfo	String	10		Purpose of cash receipt
deductionDetail	String	10		Registration method for cash receipt income deduction
userinfo	String	15		User information of cash receipt

Note. 1 – *ver ~ param3* parameters are returned same as they are sent in the request.

Note. 2 – The response is sent to *returnurl* and *statusurl* which are set in the request.

Note. 3 – *cardholder* can be different from the name printed on the card.

Note. 4 – if *statusurl* is not received properly, it is sent repeatedly 3 times, the repeat rule can be different depends on the configuration.

Note 5 – parameters related to cash receipt require a separate merchant setting.

2.2 Payer Authentication

Payer Authentication means the identity authentication of the card issuer, and after successful authentication, approval can be processed through **PAYMENT_PA** or **AUTHORIZE_PA**.

2.2.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/BasicProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/BasicProcessor.krp>

2.2.2 Request Parameters

Field Name	Type	Length	Required	Description
txntype	String	30	R	"PAYER_AUTH"
(The following is the same as the 2.1.2 request parameter)				

2.2.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"PAYER_AUTH"
ref	String	30	R	Merchant-generated unique order number or tracking number. Same value returns.
cur	String	3	R	e.g. USD, SGD, KRW...(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
email	String	50	R	Buyer Email Address
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
payerauthid	String	24	C	PAYER_AUTH Transaction id After that, use PAYMENT_PA or AUTHORIZE_PA.
rescode	String	6	R	Response Code. (Success:'0000')
resmsg	String	700		Response Message
fgkey	String	64	R	Refer to "5.1 FGKEY"
cavv	String	40		Certification result cavv
xid	String	40		Certification result xid

eci	String	28	Certification result eci
-----	--------	----	--------------------------

Note. 1 – *ver ~ param3* parameters are returned same as they are sent in the request.

Note. 2 –The response is sent to *returnurl* and *statusurl* which are set in the request.

Note. 3 – Returns *cavv*, *xid*, *eci* as the result of Payer Authentication.

Note. 4 – If the *statusurl* is not transmitted normally, it is retransmitted 3 times, and the retransmission rule may be different depending on the settings.

2.3 Performing a Sale and Authorizing a Payment with Payer Authentication

First, perform the Payer Authentication process separately. It is then used to process PAYMENT

(**PAYMENT_PA**) or AUTHORIZE (**AUTHORIZE_PA**) using the non-UI method.

AUTHORIZE_PA is charged only when purchased directly from a merchant or when **CAPTURE** is sent to Eximbay.



Authorization transactions are capped after approximately 7 to 14 days depending on the card issuer.



In the case of direct merchants, it is possible to transmit billing purchases directly from the merchant through VAN using the *tid ~ memberno* response parameter, and when using DCC service, *tid* and *memberno* are transmitted differently depending on whether the customer selects the local currency. Therefore, you must purchase the bill using the return value.

2.3.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/DirectProcessor.krp>

2.3.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"PAYMENT_PA" or "AUTHORIZE_PA"
ref	String	30	R	Merchant-generated unique order number or tracking number for each transaction

cur	String	3	R	e.g. USD, SGD, KRW...(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0) (Should be the same as the request parameter value of PAYER_AUTH)
payerauthid	String	24	R	PAYER_AUTH Transaction id (PAYER_AUTH response parameter value)
lang	String	2	R	Payment screen language (Refer to Appendix B)
returnurl	String	255		Page to be moved, after processing
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "5.1 FGKEY"
charset	String	32		Default charset – UTF-8 (Refer to Appendix C)

Note. 1 – No particular UI is provided. The refund result is returned to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

2.3.3 Response parameters

Field Name	Type	Length	Required	Description
txntype	String	30	R	"PAYMENT_PA" or "AUTHORIZE_PA"
(The following is the same as the 2.1.3 response parameter)				

2.4 Capturing an Authorization

Use to request a claim for an **AUTHORIZE** or **AUTHORIZE_PA** transaction.

2.4.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/DirectProcessor.krp>

2.4.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"CAPTURE"
ref	String	30	R	Original approved transaction ref
cur	String	3	R	Originally approved transaction currency
amt	String	15	R	Originally approved transaction amount (e.g. 1000.50, 9.15)
transid	String	24	R	Transaction ID of the original approved transaction
lang	String	2	R	Payment screen language (Refer to Appendix B)
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "5.1 FGKEY"
charset	String	32		Default charset – UTF-8 (Refer to Appendix C)

Note. 1 – No particular UI is provided. The refund result is returned to returnUrl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

Note. 4 – **Capturing an Authorization does not support capturing partial amounts.**

2.4.3 Response parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"CAPTURE"
ref	String	30	R	Merchant-generated unique order number or tracking number. Same value returns.

cur	String	3	R	e.g. USD, SGD, KRW...(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
lang	String	2	R	Payment screen language (Refer to Appendix B)
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
transid	String	24	R	Transaction id
rescode	String	6	R	Response Code. (Success:'0000', Refer to Chapter 4)
resmsg	String	700	R	Response Message
resdt	Integer	14	R	YYYYMMDDHHMMSS
allowedpvoid	String	1		Y: Partial cancellation is possible, N: Partial cancellation is not possible (default)
fgkey	String	64		Refer to "5.1 FGKEY"

2.5 statusurl

Merchant needs to send Eximbay an acknowledgement message once they have received and updated the transaction result via statusurl successfully. In order to do that the following line message need to be sent back via the same connection or printed out on the same page:

- For success : `rescode=0000&resmsg=Success`
- For fail : `rescode=(failure code)&resmsg=(failure message)`



If the acknowledgement message is not sent back to us, *statusurl* can be called multiple times. You can recognize duplicated calls if same *transid* is received. It's recommended that even if you receive duplicate call, you need to send back us the acknowledgement message.

2.6 Cash Payment Notice

It is used in **virtual account payment** methods among payment methods (Appendix D.)

Deposit Notice

Cash payment returns the payment result in 2 steps.

Step1. **Order Registration** : returnurl, statusurl

Step2. **Deposit Notice** : statusurl



The *statusurl* for notifying the order registration in step1 can be set up to not send if the merchant doesn't want to receive it.

The statusurl of Step1 and Step2 can be distinguished by "*status*" parameter.

Step1. **Order Registration** : Registered

Step2. **Deposit Notice** : Sale

In addition to the response parameters, "virtualaccountnumber" and "virtualaccountexpired" are sent when ordering a virtual account.

The parameter is the 'virtual account number and deposit validity' assigned to the customer.



Virtualaccountnumber, virtualaccountexpired

Eximbay informs the customer of the virtual account number and the deposit deadline.

3 Common Management Interfaces

3.1 Crediting a Payment

This interface is used to refund transactions of **PAYMENT** or **PAMENT_PA**, **AUTHORIZE_PA**. It usually takes 3-4 business days for acquiring bank to transfer and return money to the customer.

3.1.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/DirectProcessor.krp>

3.1.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	MerchantID provided by Eximbay
txntype	String	30	R	"REFUND"
refundtype	String	1	R	"F" : Fully, "P" : Partial
ref	String	30	R	Original transaction's order number
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount. (e.g. 1000.50, 9.15) (Do not use comma ','). Must be more than 0)
amt_taxFree	String	15	C	Non-taxable amount of refund request amount. (Do not use comma ',') Total Duty Free amount to Cancel
amt_taxable	String	15	C	Taxation amount of refund request amount (Do not use comma ',') Total taxable amount to Cancel
amt_vat	String	15	C	VAT of refund request amount (Do not use comma ',') Total VAT to Cancel
amt_serviceFee	String	15	C	Service fee of refund request amount (Do not use comma ',') Total service fee to Cancel
refundamt	String	15	R	Amount to be refunded It cannot exceed the original transaction amount. If it is empty and refundtype is set to F, the full amount of the original transaction is refunded.
balance	String	10, 2		Available balance = Transaction amount – Sum total

				refunded amounts (based on cur) Used to verify refundable balance in Eximbay for exactness. Only used when the parameter is provided.
transid	String	24	R	Original Transaction ID
refundid	String	30	R	Unique refund request id generated from merchant. Each request should have a unique value.
reason	String	255	R	Reason of refunding the transaction
lang	String	2	R	Payment screen Language (Refer to Appendix B)
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "5.1 FGKEY" (For fgkey creation, refundid need to be added)
charset	String	32		Default UTF-8
username	String	20		Account holder name (Required for refund of setlebank)
accountnumber	String	15		account number (Required for refund of setlebank)
bankcode	String	3		Bankcode (Required for refund of setlebank) Bank code for the account to be refunded (Refer to Appendix E)

Note. 1 – No particular UI is provided. The refund result is returned to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

Note. 4 – Conditions for partial refund:

- if *refundtype* is not set with value **P**, full amount will be refunded.
- You can request partial refund any number of times if the total sum of the partial request amounts are not more than the original transaction amount.

Note. 5 – Do not use *amt_taxable* if the payment method is kakaopay.

Note. 6 – When using online third-party (toss, kakaopay, payco), the *amt_taxfree*, *amt_taxable*, *amt_vat*, *amt_serviceFee* parameters are not required, and when sending parameters, they are sent to each payment company without processing them.

Note. 7 – When using naverpay point as a payment method, the values *amt_taxfree*, *amt_taxable*, *amt_vat* are required and are passed to the payment company without processing.

3.1.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	MerchantID provided by Eximbay
txntype	String	30	R	"REFUND"

ref	String	30	R	Original transaction's order number
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount. (e.g. 1000.50, 9.15)
refundamt	String	15	R	Refunded amount
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
transid	String	24	R	Original Transaction ID
refundid	String	30	R	Unique refund request id generated from merchant. Each request should have a unique value.
rescode	String	6	R	Response Code. (Success : 0000, Refer to 5.2 Response Code)
resmsg	String	700	R	Response Message
authcode	String	8		Approval Code (No value for online third-party payments)
resdt	Integer	14	R	YYYYMMDDHHMMSS (Response Date Time, Complete Date Time)
refundtransid	String	24		Refund transaction ID
fgkey	String	64	R	Refer to "5.1 FGKEY"
balance	String	15		Available balance = Transaction amount – Refund amount (based on currency <i>cur</i>) Used to verify refundable balance for exactness.

Note. 1 – *ver ~ transid* parameters are returned same as they are sent in the request.

3.2 Querying a Single Transaction

This interface is used to query a transaction detail and verify its result through API call from Merchant's Admin site. The transaction result can also be checked and verified by logging in Eximbay Admin site.

3.2.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/DirectProcessor.krp>

3.2.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	MerchantID provided by Eximbay
txntype	String	30	R	"QUERY"
keyfield	String	10	R	"REF" or "TRANSID"(default) For case of REF, the very last transaction(can be either payment or refund if they have same REF) is returned. For case of TRANSID, only its transaction is returned(TRANSID is unique for every transactions).
ref	String	30	R	Original transaction ref
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount (e.g. 1000.50, 9.15) Note : do not use comma ','
transid	String	24	C	Original transaction ID
lang	String	2	R	Payment screen Language (Refer to Appendix B)
returnurl	String	255		The page to be redirected
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "5.1 FGKEY"
charset	String	32		Default UTF-8

Note. 1 – No particular UI is provided. The query result is forwarded to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

Note. 4 – if the parameter *keyfield* is not defined, "TRANSID" is set as default. And the parameter *transid* is mandatory.

Note. 5 – The refund transaction cannot be queried by this interface, the refund status can be checked with the response parameter *balance* by querying its original transaction. If balance is 0, you can consider the transaction has been refunded fully.

3.2.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"QUERY"
ref	String	30	R	Original transaction ref
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount (e.g. 1000.50, 9.15)
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
status	String	10	R	Payment Completed : SALE (Payment confirmed) Authorization : AUTH (Payment not confirmed, confirmed after Capture) Order Registration : REGISTERED (Payment not confirmed, confirmed after Deposit Notice) No Transaction : NONE
transid	String	24	R	Original transaction ID
rescode	String	6	R	Response Code. (Success:'0000'; Refer to Chapter 4)
resmsg	String	700	R	Response Message
authcode	String	8		Approval Code (Only provided for credit card payment)
resdt	Integer	14	R	YYYYMMDDHHMMSS (Transaction Date)
fgkey	String	64	R	Refer to "5.1 FGKEY"
cardno1	Integer	4		First 4 digit of card number
cardno4	Integer	4		Last 4 digit of card number
balance	String	15		Balance(Refundable Amount) = Transaction Amount – Sum Total Refunded Amount Refer to the next table for possible values.
paymethod	String	4		Payment Method Code (Refer to Appendix C)

Note. 1 – Following values are returned for *balance* depending on the status

Transaction category	status	balance	Description
AUTHORIZE AUTHORIZE_PA	AUTH	Same with amt	

AUTH_REVERSAL	AUTH	0	Reversal transaction
PAYMENT CAPTURE PAYMENT_PA	SALE	Transaction Amount – Sum total refunded amounts	Available Refundable Amount

3.3 Querying a Cash Receipt Transaction

Check the cash receipt issuance details.

3.3.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/DirectProcessor.krp>

3.3.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"230"
mid	String	10	R	MerchantID provided by Eximbay
txntype	String	30	R	"CASHRECEIPT"
transid	String	24	R	Transaction ID of the original approved transaction
fgkey	String	64	R	Refer to "5.1 FGKEY"
charset	String	32		Default charset – UTF-8

Note. 1 – No particular UI is provided. The refund result is returned to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

3.3.3 Response Parameters

Field Name	Type	Length	Required	Description
product	String	80	R	Product name
approvalCode	String	9		Approval Code
buyerEmail	String	60	R	Email
uniqueValue	String	80	R	Unique number
buyerName	String	30	R	Buyer
transactionId	String	24	R	Original transaction ID
vatAmount	String	18	R	VAT
responseCode	String	10		Response Code
transactionType	String	4	R	Transaction type
totalAmount	String	18	R	Payment amount
buyerContactNo	String	40	R	Contact number

supplyAmount	String	18	R	Supply amount
serviceAmount	String	18	R	Service amount
fgkey	String	64	R	Refer to "5.1 FGKEY"
merchantId	String	10	R	MerchantID provided by Eximbay
approvalDt	String			Approval date
bizId	String	10	R	Business number
issuingType	String	1	R	Purpose of Issue 1: For income deductions 2 : For proof of expenditure
responseMessage	String	255		Response message
status	String	4	R	Status. Refer to "Appendix F"

4. Secondary Partner Preferential Fee Differential Settlement Process

This service is a PG secondary partner preferential fee settlement process and is only applied to **domestic credit card payments**. Existing "Performing a Sale" and "Crediting a Payment" requests require additional parameters.

4.1 Performing a Sale (PG Secondary Partner Additional Parameters)

When requesting approval, additional parameters defined below are required.

4.1.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/BasicProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/BasicProcessor.krp>

4.1.2 Request Parameters

Field Name	Type	Length	Required	Description
item_#_subbizno	String	10	R	Subcontractor's Business Number

(The following is same as the **2.1.2 request parameter**)

Note. 1 –The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document.

Note. 2 – # is a positive integer starting from 0.

Note. 3 – The parameter *item_#_subbizno* shares the same '#' as the value of '*item_#_product*', '*item_#_quantity*', and '*item_#_unitPrice*', which represents product information when sending a payment request. when requesting the payment of two or more sub-business products together, the data should be set up in the following ways:

amt=50000

item_0_product=productofA

item_0_quantity=1

item_0_unitPrice=36000

item_0_subbizno=1138637262

item_1_product=productofB

item_1_quantity=2

item_1_unitPrice=7000

item_1_subbizno=1178300669

4.1.3 Response Parameters

Field Name	Type	Length	Required	Description
(The following is same as the 2.1.3 response parameter)				

Note. 1 –The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document.

Note. 2 – # is a positive integer starting from 0.

4.2 Crediting a Payment (Additional Parameters)

When requesting cancellation, you must add the following parameters:

4.2.1 Request URL

- Test <https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp>
- Production <https://secureapi.eximbay.com/Gateway/DirectProcessor.krp>

4.2.2 Request Parameters

Field Name	Type	Length	Required	Description
item_#_subbizno	String	10	R	Subcontractor's Business Number
item_#_unitPrice	String	15	R	Amount requested for refund by item
(Same as other Crediting a Payment cancellation request parameters)				

Note. 1 –The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document.

Note. 2 – # is a positive integer starting from 0.

4.2.3 Response Parameters

Field Name	Type	Length	Required	Description
(Same as other Crediting a Payment cancellation response parameters)				

5 References

5.1 FGKEY

FGKEY is used to authenticate and validate parameters transferred between the merchant and Eximbay.

5.1.1 How it works

The SHA-256 hashing is a specific way of encrypting information to make it unreadable. You can generate your *fgkey* value by encrypting a string composed of specific parameters and send it in your request. If the *fgkey* value that you generated matches with the value we generated, your request is considered as valid, otherwise it is rejected. In the same way you can verify against the *fgkey* field provided in Eximbay response.

5.1.2 Setting up the SHA-256 hashing

A secret key is required to generate the *fgkey* and it is assigned to you by Exmbay. You must keep it confidential.

5.1.3 Hash via the merchant

You should use the following steps to generate your *fgkey* value:

- A. You need to sort all request/response parameters in alphabetical order
- B. Put the secretkey and A's data connecting together with "?" symbol in between
- C. Encrypt B's result with SHA256 Hashing function

Example:

- secretkey
F6DCE41DA82064F478B934663FD2A07E
- Eximbay link buffer
-Request
A = `sort("ver=230&mid=1234567890&ref=A1234&cur=USD&amt=1.0.....")`
-Response
A = `sort("ver=230&mid=1234567890&ref=A1234&cur=USD&amt=1.0&rescode=0000.....")`
- SHA-256 encryption
B = `"F6DCE41DA82064F478B934663FD2A07E?A"`
C = `SHA256(B.getBytes("UTF-8"))`

Note. 1 – You need to set character set as UTF-8 when converting to byte data for SHA256 encryption.

- Set the generated hash value to fgkey parameter
fgkey=C

5.2 Response Code

When you receive the response parameter *rescode* with value 0000, the transaction is successful, otherwise it is a failure. For the failure reason, you may refer to the response parameter *resmsg*.

We cannot provide all the failure codes because each acquiring bank has different codes. But we may provide Eximbay system failure codes if merchants require.

5.3 AES 256 Encryption

The confidential parameters like cardno(credit card number) and expirydt(expiration date) are returned in the response as encrypted by AES 256 encryption algorithm. If you want to decrypt the parameters, you can do as following:

- A. Encryption Algorithm : AES256
- B. Operation mode : CBC
- C. Padding : PKCS5Padding
- D. IV Key : Eximbay_AES_256_
- E. Round Key : Eximbay allotment (Note: Round Key is provided by Eximbay)

5.4 About app-to-app

Eximbay is based on the web. In the case of merchants who use EXIMBAY domestic payment in the merchant app, additional work related to calling the payment company's app is required.

5.4.1 IOS

1. Registration of URL Scheme

It is necessary to register an external URL scheme(LSApplicationQueriesSchemes) as follows for running a 3rd party app.

```
<key>LSApplicationQueriesSchemes</key>
<array>
    <string>ispmobile</string> <!--ISP Mobile-->
```

```

<string>lotteappcard</string> <!--LOTTE Card-AppCard-- >
<string>mpocket.online.ansimclick</string> <!--SAMSUNG Card-AppCard-->
<string>shinhan-sr-ansimclick</string> <!--SHINHAN Card-AppCard-->
<string>hdcardappcardansimclick</string> <!--HYUNDAI Card-AppCard-->
<string>nhallonepayansimclick</string> <!--NONGHYUP Card All-in-one-Pay -->
<string>cloudpay</string> <!--HANA Card-AppCard -->
<string>citimobileapp</string> <!--CITI Mobile-AppCard -->
<string>kakaotalk</string> <!-- KAKAOPAY -->
<string>payco</string> <!-- PAYCO -->
<string>smilepayapp</string> <!-- SmilePay -->
<string>supertoss</string> <!-- TOSS -->
<string>naversearchthirdlogin</string> <!--NAVER Pay -->
</array>

```

2. Setting for App Transport Security

ATS(App Transport Security) needs to be set up as follows to allow security restrictions for HTTP requests entering the web view.

```

<key>NSAppTransportSecurity</key>
<dict>
  <key>NSAllowsArbitraryLoadsInWebContent</key>
  <true/>
  <key>NSAllowsArbitraryLoads</key>
  <true/>
</dict>

```

5.4.2 ANDROID

Additional work related to calling the payment company's app is required on WebView, `shouldOverrideUrlLoading`.

```

@Override
public boolean shouldOverrideUrlLoading(WebView view, String url) {
    if (!URLUtil.isNetworkUrl(url) && !URLUtil.isJavaScriptUrl(url)) {
        final Uri uri;
        Intent intent = null;

        try {
            uri = Uri.parse(url);
            intent = Intent.parseUri(url, Intent.URI_INTENT_SCHEME);
        } catch (Exception e) {

```

```

        return false;
    }

    if ("intent".equals(uri.getScheme())) {
        try {
            Log.d("LOG", "intent startActivity");
            startActivity(intent);
            return true;
        } catch (ActivityNotFoundException e) {
            final String packageName = intent.getPackageName();
            Log.d("LOG", "ActivityNotFoundException packageName : " + packageName);
            if (!TextUtils.isEmpty(packageName)) {
                startActivity(new Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=" + packageName)));
                return true;
            }
        }
    }
    } else if ("supertoss".equals(uri.getScheme())) { //TOSS
        try {
            Log.d("LOG", "TOSS startActivity");
            startActivity(new Intent(Intent.ACTION_VIEW, uri));
            return true;
        } catch (ActivityNotFoundException e) {
            Log.d("LOG", "ActivityNotFoundException TOSS");
            startActivity(new Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=viva.republica.toss")));
            return true;
        } catch (Exception e) {
            return false;
        }
    }
    } else if ("nidlogin".equals(uri.getScheme())) { //NAVERPAY
        try {
            Log.d("LOG", "NAVERPAY startActivity");
            startActivity(new Intent(Intent.ACTION_VIEW, uri));
            return true;
        } catch (ActivityNotFoundException e) {
            Log.d("LOG", "ActivityNotFoundException NAVERPAY");

```

```

        startActivity(new Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=com.nhn.android.search")));
        return true;
    } catch (Exception e) {
        return false;
    }
} else {
    try {
        Log.d("LOG", "else startActivity");
        startActivity(new Intent(Intent.ACTION_VIEW, uri));
        return true;
    } catch (ActivityNotFoundException e) {
        final String packageName = intent.getPackageName();
        Log.d("LOG", "4091_else startActivity ActivityNotFoundException
packageName : " + packageName);
        if (!TextUtils.isEmpty(packageName)) {
            startActivity(new Intent(Intent.ACTION_VIEW, Uri.parse("market://details?id="
+ packageName)));
            return true;
        }
    } catch (Exception e) {
        return false;
    }
}
return false;
}

```

Appendix A Supported Currencies

A.1 Request Currencies (cur)

Currency Code	Currency ID(ISO4217)	Currency Name	Minor Unit
KRW	410	Korea Won	0
USD	840	US Dollar	2
EUR	978	Euro	2
GBP	826	Pounds Sterling	2
JPY	392	Japan Yen	0
THB	764	Thailand Baht	2
SGD	702	Singapore Dollar	2
RUB	643	Russian Ruble	2
HKD	344	Hong Kong Dollars	2
CAD	124	Canadian Dollars	2
AUD	036	Australian Dollars	2

Appendix B Supported Languages

Code	Language
KR	Korean

Appendix C Payment Methods

(paymethod)

paymethod	Payment Method Name
P000	CreditCard
P110	BC Card
P111	KB Card
P112	HANA Card (before it was Foreign Exchange)
P113	SAMSUNG Card
P114	SHINHAN Card
P115	HYUNDAI Card
P116	LOTTE Card
P117	NONGHYUP Card
P119	CITI Card
P120	WOORI Card
P121	SUHYUP Card
P122	JEJU Card
P123	JEONBUK Card
P124	KWANGJU Card
P125	KAKAO Bank
P126	K Bank
P127	MireaAssetDaewoo
P128	Kona Card
P301	BankPay
P302	Kakaopay
P303	TOSS
P304	Payco
P305	Virtual Account
P306	SmilePay
P105	NAVER Pay(Card and Point)
P307	NAVER Pay(Card)
P308	NAVER Pay(Point)

Appendix D State, Province and Territory Codes

D.1 For the United States and Canada

United States Postal Service(USPS) Abbreviations

State	Code	State	Code
Alabama	AL	Montana	MT
Alaska	AK	Nebraska	NE
American Samoa	AS	Nevada	NV
Arizona	AZ	New Hampshire	NH
Arkansas	AR	New Jersey	NJ
California	CA	New Mexico	NM
Colorado	CO	New York	NY
Connecticut	CT	North Carolina	NC
Delaware	DE	North Dakota	ND
District of Columbia	DC	Northern Mariana Islands	MP
Federated States of Micronesia	FM	Ohio	OH
Florida	FL	Oklahoma	OK
Georgia	GA	Oregon	OR
Guam	GU	Palau	PW
Hawaii	HI	Pennsylvania	PA
Idaho	ID	Puerto Rico	PR
Illinois	IL	Rhode Island	RI
Indiana	IN	South Carolina	SC
Iowa	IA	South Dakota	SD
Kansas	KS	Tennessee	TN
Kentucky	KY	Texas	TX
Louisiana	LA	Utah	UT
Maine	ME	Vermont	VT
Marshall Islands	MH	Virgin Islands	VI
Maryland	MD	Virginia	VA
Massachusetts	MA	Washington	WA
Michigan	MI	West Virginia	WV
Minnesota	MN	Wisconsin	WI
Mississippi	MS	Wyoming	WY
Missouri	MO		

U.S Military Address Format

Line	Description	Examples
Address	Street address Post office box number	1775 John Paul Jones BLVD PO Box 405

	Postal service center number Unit and/or box number	4PSC 467 Box 291 Unit 30001 Unit 62001 Box 426
City	APO (Army or Air Force Post Office) or FPO (Fleet Post Office for the Navy, Marine Corps, or Coast Guard)	FPO AE 09501-4665 FPO AA 34093-2329 FPO AP 96349-1100
State	AE if customer is in Africa, Canada, Europe, or the Middle East AP if customer is in Pacific AA if customer is elsewhere in the Americas	
ZIP	Postal code of the customer's location	
Country	Country of the customer's location or US if customer is not currently assigned to another country (FPO)	

Canadian Province or Territory Abbreviations

Province or Territory(English)	Description
Alberta	AB
British Columbia	BC
Manitoba	MB
New Brunswick	NB
Newfoundland and Labrador	NL
Northwest Territories	NT
Nova Scotia	NS
Nunavut	NU
Ontario	ON
Prince Edward Island	PE
Quebec	QC
Saskatchewan	SK
Yukon	YT

Appendix E Bank Code

Bankcode	Bank Name
001	Bank of Korea
002	KDB
003	IBK
004	KB Bank
005	KEB hana
007	Suhyup
008	Koreaexim
011	NH Bank
012	NongHyup
020	Woori
023	Standard Chartered
027	Citi Bank
031	DGB
032	BNK
034	KJ Bank
035	Jeju Bank
037	JB Bank
039	KN Bank
045	KFCC
048	NACUFOK
050	Mutual Savings Bank
052	Morgan Stanley Bank
054	HSBC Bank
055	Deutsche Bank
071	Epost Bank
081	KEB Hana
088	Shinhan
089	K Bank
090	Kakao Bank

Appendix F Cash receipt issuance status

status	설명
CA00	Request Approval (for issuing cash receipts)
CA01	Approval Processing Completed (National Tax Service issue successfully. .)
CA02	Approval Processing failed (Failed to issue NTS)
CA03	Cancellation Request (Request to cancel issuing cash receipts)
CA04	Cancellation Processing Completed (NTS issue cancellation completed.)
CA05	Cancellation processing failed (NTS cancellation failed.)
CA06	Issue Changes. (Transaction in progress of issuance of changes)