



DOCUMENT OF GUIDELINE FOR CONNECTING TO MEGAPAY PAYMENT GATEWAY

Hà Nội Feb-2019



1 Modification history

Date	Author	Version	Content
01/12/2018	Bùi Văn Luận	1.0	Create new document
21/03/2019	Nguyễn Minh Huân	1.1	Modify some required/optional fields
03/04/2019	Nguyễn Minh Huân	1.2	Remove the error codes which is not used
05/04/2019	Bùi Văn Luận	1.2.1	- Add error code OR_140, DC_122, DC_123 ,DC_124, DC_125, VA_112, VA_113 - Remove param: encodeKey
16/04/2019	Bùi Văn Luận	1.2.2	- Add failure error code: PG_ER2, PG_ER3, PG_ER4, PG_ER5 - Input field named “description” for mandatory in payment message
18/04/2019	Nguyễn Minh Huân	1.2.2	Change userIP from optional status to mandatory status
16/05/2019	Bùi Văn Luận	1.2.3	- Change from Mandatory to option: buyerCountry, buyerCity, buyerAddr - Add error code: OR_147 – description is invalid
23/05/2019	Bùi Văn Luận	1.2.3	Description Deposit code payment method
11/06/2019	Bùi Văn Luận	1.2.4	Add more error code (PG_ER7 - PG_ER30)
20/08/2019	Bùi Văn Luận	1.2.5	- Add the error code OR_122, OR_148 - Add resultCd in the formula of generating merchant token - Add timeStamp and merchantToken in Inquiry response - Change format of Refund response
17/09/2019	Bùi Văn Luận	1.2.6	- Added payment by tokenization - Change the formula of generating merchantToken for payment if tokenization was used: Added field named payToken - Change the formula of generating merchantToken for responding payment, IPN, Inquiry transaction if tokenization was used: Added field named payToken - Added error code IC_122, IC_123, IC_124, OR_150, OR_151, OR_152, OR_153 - Inquiry transaction: exclude all NULL data field from response string - Add bankCode field into Payment method
15/11/2019	Bùi Văn Luận	1.2.7	Add payment with tokenization directly API DC_126
07/01/2020	Bùi Văn Luận	1.2.7	- Return issueBankName, cardType if pay with international card - Add error code: DC_125
01/04/2020	Nguyễn Thanh Tùng	1.2.8	- Add parameter cardTypeValue, cardTypeToken if pay with international card - Add error code: IC_127, IC_128, IC_129
04/05/2020	Trần Tuấn Đạt	1.2.9	Modify 4.1 and 4.2: payment ATM token
26/06/2020	Trần Tuấn Đạt	1.3.0	Modify 4.2: validate merchantToken from merchant side
28/07/2020	Nguyễn Thanh Tùng	1.31	Add paytpe EW ewallet when payment Change status when get trx status Add the subappid field in section 4.1 for payment request
23/10/2020	Bui Van Luan	1.3.2	Add integration flow
11/12/2020	Duong Van Hung	1.3.3	Add new payment method: Momo ewallet. Add more error code (PG_ER42, PG_ER43)
15/12/2020	Nguyen Thanh Tung	1.3.4	Add more error code for ewallet(from number 152 to 182)
16/12/2020	Duong Van Hung	1.3.4	Add new payment method: Installment
11/03/2021	Tran Trong Dat	1.3.5	Add payOption = PAY_CREATE_TOKEN
18/03/2021	Tran Trong Dat	1.3.5	Add more error code OR_164, OR_165



19/05/2021	Duong Van Hung	1.3.5	<ul style="list-style-type: none">- Delete parameter in request: userFee, goodsAmount- Add more parameter in response: userFee- Add more bank code of installment: FECM: FE credit HMCM: Home credit
06/08/2021	Duong Van Hung	1.3.6	<ul style="list-style-type: none">- Add delete token API- Add payOption in response (IPN) and Inquiry- Add a signature with the merchant using the user to bear the fee in the response- Add cardEnroll3DS param in the response
30/08/2021	Nguyen Thanh Tung	1.3.7	<ul style="list-style-type: none">Add new payment method: Moca ewallet.Add more error code for moca (MC_1 -> MC_19)
06/10/2021	Duong Van Hung	1.3.8	<ul style="list-style-type: none">- Add more error code: OR_152 → OR_173, DC_127 → DC_130, PG_ER31 → PG_ER32
13/10/2021	Dinh Quoc Cuong	1.3.8	<ul style="list-style-type: none">- Add more error code: IC_130 → IC_139
20/10/2021	Dinh Quoc Cuong	1.3.8	<ul style="list-style-type: none">- Add more error code: IC_140 → IC_145, OR_174
08/11/2021	Dinh Quoc Cuong	1.3.8	<ul style="list-style-type: none">- Add more error code: IC_146, DC_131 → DC_135, OR_175
14/12/2021	Duong Van Hung	1.3.9	<ul style="list-style-type: none">- Add Installment payment API:<ul style="list-style-type: none">+ Section 5.1 adds the termIs field, edits the bankCode, amount . field information+ Add API to get installment information at section 11+ Add error code IS_001- Add field: subMerId
18/01/2022	Duong Van Hung	1.4.0	<ul style="list-style-type: none">- Add API check connection information for merchant- Callback, IPN add field: icIssue
03/03/2022	Duong Van Hung	1.4.1	<ul style="list-style-type: none">- Add more error code: 00_005 (Transaction has been map with Dcode account success, but end user has not paid)- Add new payment method: VNPAYQR- Add a new bank CFCM (VietCredit Finance Joint Stock Company)- Change name: Virtual Account --> Deposit code- Add new ewallet : ViettelPay, ShopeePay
04/07/2022	Duong Van Hung	1.4.2	<ul style="list-style-type: none">- Add param in request: site, soldToCode, productCategory- Request add list object: productList (Section 5)- Refund add list object: refundData (Section 8)- Add more error code of Viettel Money: VT_01 → VT_13 (Section 15)- Add 2 step payment with international card feature (Section 5, 6, 9, and 14)- Change domain on sandbox: remove port- Add new error code: IC_147, IC_148, IC_149- Add new bank: KEHB, KEHC



2 Table of contents

1	Modification history.....	2
2	Table of contents.....	4
3	Integration flow.....	6
3.1	General payment	6
3.2	Payment & token generation.....	7
3.3	Pay with token.....	8
4	Test system's information.....	10
4.1	Merchant's information	10
4.2	Testing card list.....	10
5	Payment.....	11
5.1	Request.....	11
5.2	Response	18
5.2.1	Redirect	19
5.2.2	Send notification to Merchant via API (IPN - Instant Payment Notification).....	23
6	Pay with Tokenization (Direct API Payment)	27
6.1	Request.....	27
6.2	Response	28
7	Pay by Deposit code:.....	30
8	Refund transaction	32
8.1	Request.....	32
8.2	Response	34
9	Inquiry transaction status	35
9.1	Request.....	35
9.2	Response	35
10	Installment cancel (IS cancel)	41
10.1	Request	41
10.2	Response.....	42
11	API to get installment information.....	43
11.1	Integration flow	43
11.2	Request	44
11.3	Response.....	44
12	Delete unused token.....	46
12.1	Request	46
12.2	Response.....	46
13	API to check connection information	47
13.1	Request	47

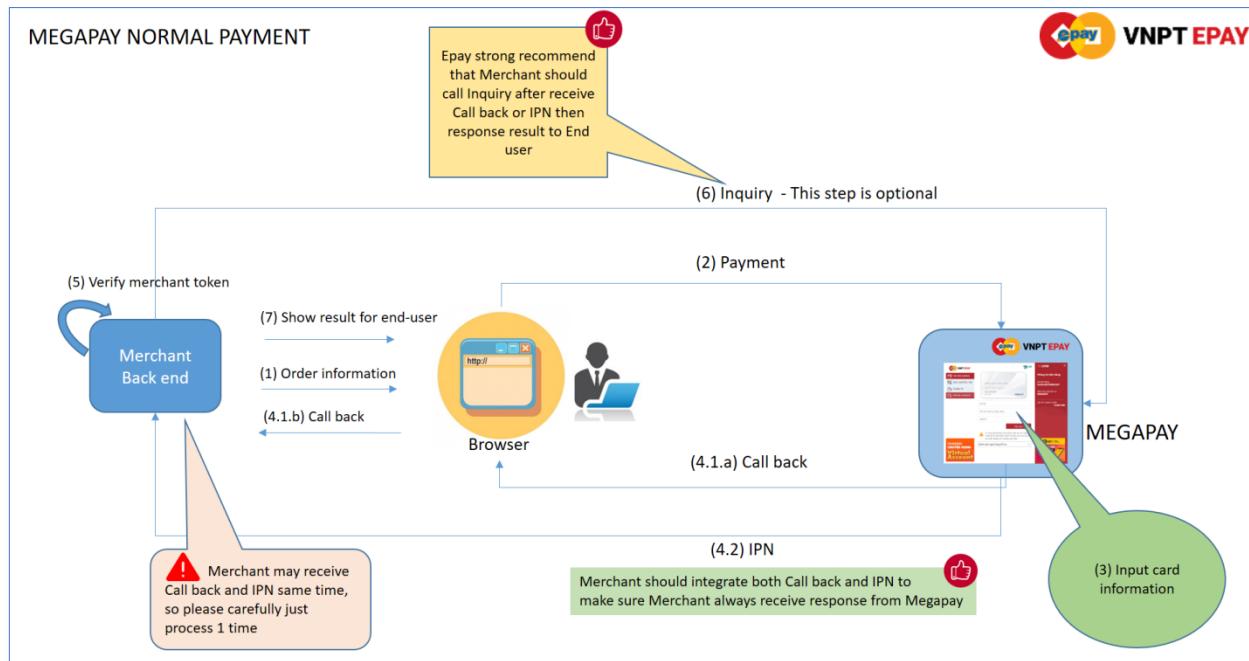


13.2	Response.....	48
14	Capture Request API.....	49
14.1	Integration flow	49
14.2	Request	51
14.3	Response.....	51
15	Appendix.....	52
15.1	Payment method list	52
15.2	Error code list	52
15.3	List bank	63
16	Payment screen.....	64
16.1	Payment window of ATM card:	64
16.2	Payment window of International card:	65
17	Payment guide with ZaloPay	66
17.1	Payment window	66
17.2	Download ZaloPay app in Sandbox version	67
17.3	Use ZaloPay app (Sandbox version) & scan QR to pay.....	67
18	Payment guide with Momo	69
18.1	Payment window	69
18.2	Download Momo app in Sandbox version	69
18.3	Use Momo app (Sandbox version) & scan QR to pay	70
19	Payment guide with Moca.....	72
19.1	Payment window	72
19.2	Download Moca app in Sandbox version.....	73
19.3	Use Moca app (Sandbox version) & scan QR to pay.....	74
20	Payment guide with VNPAYQR	76
20.1	Payment window	76
20.2	Download MSB mBank app in Sandbox version.....	77
20.3	Use MSB mBank app (Sandbox version) & scan QR to pay	77
21	Payment guide with ShopeePay	79
21.1	Payment window	79
21.2	Download Shopeepay app (sandbox version)	80
21.3	Use ShopeePay app (Sandbox version) payment with PC	82
21.4	Use ShopeePay app (Sandbox version) payment with Mobile	84
22	Payment guide with ViettelPay.....	85
22.1	Payment window	85



3 Integration flow

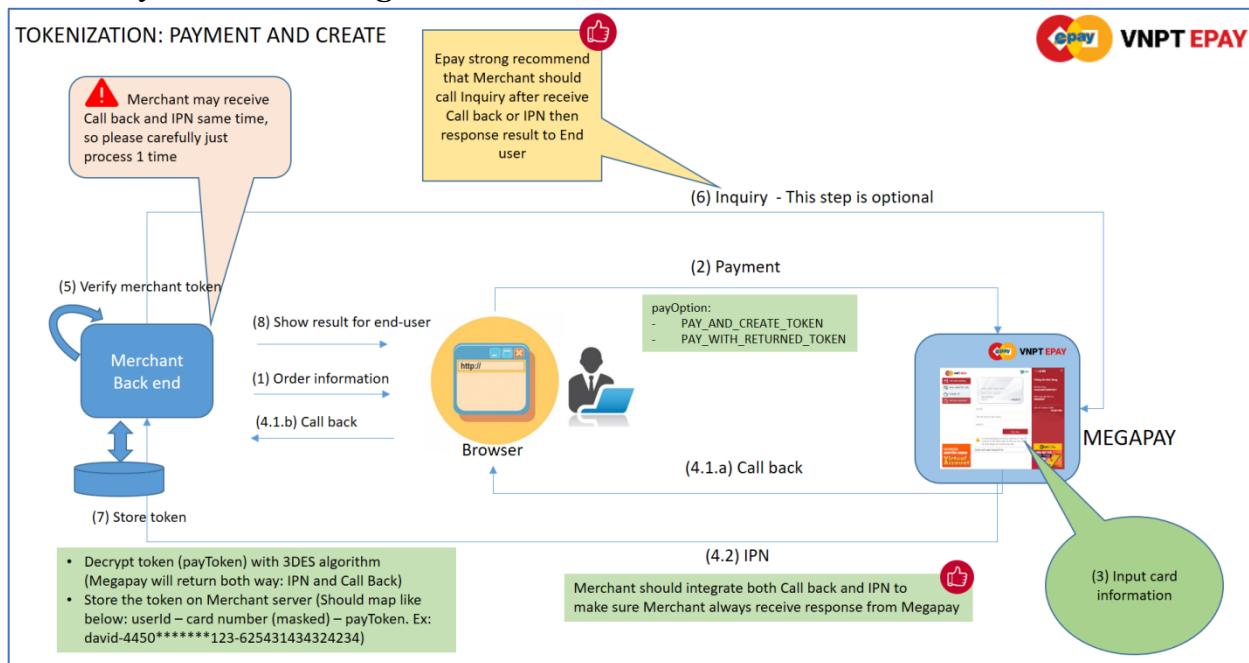
3.1 General payment



Step	Description
1	Order info is displayed on Merchant's front end
2	End user clicks "Payment" on Merchant website Merchant redirects user to Megapay's gateway to input card info
3	User inputs card information
4.1.a	Megapay returns payment result to Merchant through callback (callBackUrl Parameter)
4.1.b	Front-end transfers call-back result to Merchant's back end
4.2	Megapay returns payment result to Merchant via IPN (notiUrl Parameter) Note: Megapay may return transaction result through callback and IPN same time
5	Merchant will verify Merchant's token returned by Megapay, which is to make sure that the result returned by Megapay cannot be altered through transmission. Merchant generates merchant token as per formular prescribed in the documentation then compares with merchant token returned by Megapay in merchantToken parameter If the data matches: Proceed to step 6 (Merchant can bypass this step by going straight to step 7) If the data does not match: Proceed to step 6

6	Check transaction's final status
7	<p>Upon receipt of transaction's final status:</p> <ul style="list-style-type: none"> If the status is successful, merchant will return result to end user If the status is failure, merchant will notify user with error notice If the status is pending (transaction still in processing), merchant will call transaction Inquiry function until the final status is updated by Megapay

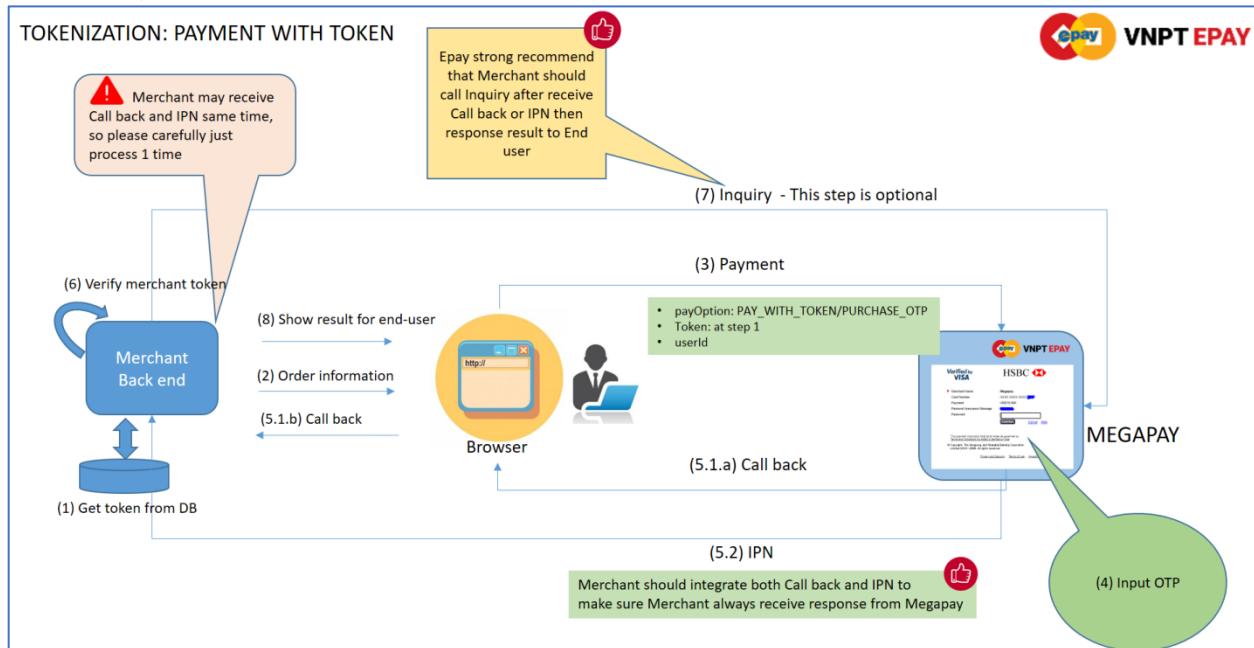
3.2 Payment & token generation



Step	Description
1	Order info is displayed on Merchant's front end
2	End user clicks "Payment" on Merchant website Merchant redirects user to Megapay's gateway to input card info
3	User inputs card information
4.1.a	Megapay returns payment result to Merchant through callback (callBackUrl Parameter)
4.1.b	Front-end transfers call-back result to Merchant's back end
4.2	Megapay returns payment result to Merchant via IPN (notiUrl Parameter)
5	Merchant will verify Merchant's token returned by Megapay, which is to make sure that the result returned by Megapay cannot be altered through transmission. Merchant generates merchant token as per formular prescribed in the

	<p>documentation then compares with merchant token returned by Megapay in merchantToken parameter</p> <ul style="list-style-type: none"> • If the data matches: Proceed to step 6 (Merchant can bypass this step by going straight to step 7) • If the data does not match: Proceed to step 6
6	<p>Merchant checks transaction's final status, which is optional but recommended</p>
7	<p>Upon receipt of transaction's final status:</p> <ul style="list-style-type: none"> • If the status is successful, merchant will save token, card number and user info. • If the status is failure, merchant will notify user with error notice • If the status is pending (transaction still in processing), merchant will process the transaction according to the final status (successful/failure) as above.
8	<p>Upon receipt of transaction's final status:</p> <ul style="list-style-type: none"> • If the status is successful, merchant will return result to end user • If the status is failure, merchant will notify user with error notice • If the status is pending (transaction still in processing), merchant will call transaction Inquiry function until the final status is updated by Megapay then follow the steps above.

3.3 Pay with token





Step	Description
1	Merchant acquires the token information stored in Payment with Token step then transfers to Megapay for making payment This token substitutes user's input information.
2	Order info is displayed on Merchant's front end
3	End user clicks "Payment" on Merchant website Merchant redirects user to Megapay's gateway to input card info
4	User inputs card information
5.1.a	Megapay returns payment result to Merchant through callback (callBackUrl Parameter)
5.1.b	Front-end transfers call-back result to Merchant's back end
5.2	Megapay returns payment result to Merchant via IPN (notiUrl Parameter)
6	Merchant will verify Merchant's token returned by Megapay, which is to make sure that the result returned by Megapay cannot be altered through transmission. Merchant generates merchant token as per formular prescribed in the documentation then compares with merchant token returned by Megapay in merchantToken parameter <ul style="list-style-type: none">• If the data matches: Proceed to step 6 (Merchant can bypass this step by going straight to step 7)• If the data does not match: Proceed to step 6
7	Merchant checks transaction's final status, which is optional but recommended
8	Upon receipt of transaction's final status: <ul style="list-style-type: none">• If the status is successful, update the status and return result to user• If the status is failure, update the status to failure according to the error code returned by Megapay and notify user with an error notice.• If the status is pending (transaction still in processing), merchant will call transaction Inquiry function until the final status is updated by Megapay then follow the steps above.



4 Test system's information

4.1 Merchant's information

- Merchant ID (merId): EPAY000001
- Checksum key (encodeKey):
rf8whwaejNhJiQG2bsFubSzccfRc/iRYyGUUn6SPmT6y/L7A2XAbbu9y4GvCoSTOTpv
JykFi6b1G0crU8et2O0Q==

4.2 Testing card list

ATM

Bank	Card number	Card holder	Effective /Expried date	OTP/CVV
NAPAS	9704 0000 0000 0018	NGUYEN VAN A	03/07	otp
AB Bank	9704250000000001	NGUYEN VAN A	12/08	123456

Visa/Master/JCB

Bank	Card number	Expired date	CVV	3DS pass word
Visa card	445653 00 0000 1005	11/23	123	No need pass word
Master card	520000 00 0000 1005	11/23	123	No need pass word
JCB	333700 00 0000 0008	11/23	123	No need pass word
Visa card	445653 00 0000 1096	11/23	123	1234
Master card	520000 00 0000 1096	11/23	123	1234
JCB	333700 00 0020 0004	11/23	123	1234

ViettelPay

Viettelpay	User	Pass
	0983725525	111111



5 Payment

5.1 Request

Step 1: Build a form base on the list of Megapay's parameters

Note: Form's properties must be exactly as this:

```
<form id="megapayForm" name="megapayForm" method="POST">
```

Insert Megapay's CSS file and JS file into the head of HTML page.

Test server's information"

- File JS: https://sandbox.megapay.vn/pg_was/js/payment/layer/paymentClient.js
- File CSS:
https://sandbox.megapay.vn/pg_was/css/payment/layer/paymentClient.css
- Domain: <https://sandbox.megapay.vn>

Step 2: Initialize the payment page:

After creating the payment form filled with required parameters, merchant calls "openPayment" function in Megapay's JS file to open Megapay's payment window.

Input of "openPayment" function is: openPayment(1, domain);

In which the domain will be sent to merchant separately. For testing system, the domain is provided at section 5.1.

Special note: In case of calling the payment page in "webview" from a mobile application (Android/IOS), please configure to accept cookies to ensure successful credit card transactions. (required by some card issuers' 3DS authentication pages)



```
Sample code (Android)
if (android.os.Build.VERSION.SDK_INT >= 21) {
    CookieManager.getInstance().setAcceptThirdPartyCookies(mWebView, true);
} else {
    CookieManager.getInstance().setAcceptCookie(true);
}
```



List of parameters:

Field name	Description	Required	Datatype
merId	Unique ID for Merchant Sample: [String] "MEGAPAYTES"	Yes	String(10) - Fixed
currency	Currency of each country Only allow "VND" now	Yes	String(3) – Fixed
amount	Amount of money for payment Sample: [String] "500000" 10,000 <= amount < 2,147,483,647  Note: Do not include user fee since it's calculated on our system if requested to configure before going live.	Yes	String (12)
invoiceNo	Invoice number, order number on merchant's system Need to be unique on Megapay system	Yes	String (40)
goodsNm	Good's name	Yes	String(200)
payType	Payment method: <ul style="list-style-type: none"> - IC : Credit card (Visa/master/JCB...) - DC : Domestic card (ATM) - EW: Ewallet (Zalopay, Momo, Moca, ShopeePay, Viettelpay) - VA : Deposit code - IS: Installment - QR: VNPAYQR - NO: Don't select payment method (Users will select when entering megapay's payment page) 	Yes	String(2)
cardTypeValue	- Card type. - Only apply for International payment(payType = IC), the value as below: <ul style="list-style-type: none"> - visa: Visa card - mastercard: Master Card - jcb: JCB 	No	String(30)
buyerFirstNm	Buyer's first name <ul style="list-style-type: none"> - Pay by credit card (IC): Required 	No	String(30)



	<ul style="list-style-type: none"> - Installment payment (IS): Required - Pay by ATM, EW, VA, QR: Optional <p>Note: Including symbol “_” is not allowed</p>		
buyerLastNm	<p>Buyer's last name</p> <ul style="list-style-type: none"> - Pay by credit card (IC): Required - Installment payment (IS): Required - Pay by ATM, EW, VA, QR: Optional <p>Note: Including symbol “_” is not allowed</p>	No	String(30)
buyerPhone	Buyer's phone number	No	String(15)
buyerEmail	<p>Buyer's Email</p> <ul style="list-style-type: none"> - Pay by credit card (IC): Required - Installment payment (IS): Required - Pay by ATM, EW, VA, QR: Optional 	Yes	String(40)
buyerAddr	Buyer's address	No	String(60)
buyerCity	Buyer's city	No	String(50)
buyerState	<p>Buyer's state</p> <p>This field is mandatory in case of the country which was sent by merchant (buyerCountry) is United State (US) or Canada (CA)</p> <ul style="list-style-type: none"> - State/District code of United State and Canada referenced by the list below: http://apps.cybersource.com/library/documentation/sbc/quickref/states_and_provinces.pdf 	No	String(20)
buyerPostCd	<p>Buyer's post code</p> <p>This field is mandatory in case of the country which was sent by merchant (buyerCountry) is United State (US) or Canada (CA):</p> <p>When the billing country is the U.S., the 9-digit postal code must follow this format: [5 digits][dash][4 digits]</p> <p>Example 12345-6789</p> <p>When the billing country is Canada, the 6-digit postal code must follow this format: [alpha][numeric][alpha][space] [numeric][alpha][numeric]</p> <p>Example A1B 2C3</p> <p>For the rest of the world countries, the maximum length is 10.</p>	No	AlphaNumeric (10)



buyerCountry	Buyer's country List of country code referenced by the link below: http://apps.cybersource.com/library/documentation/sbc/quickref/countries_alpha_list.pdf	No	String(2)
receiverLastNm	Receiver's last name	No	String(30)
receiverFirstNm	Receiver's first name	No	String(30)
receiverPhone	Receiver's phone number	No	String(15)
receiverAddr	Receiver's address	No	String(100)
receiverCity	Receiver's city	No	String(50)
receiverState	Receiver's district	No	String(50)
receiverPostCd	Receiver's post code	No	String(10)
receiverCountry	Receiver's country	No	String(10)
callBackUrl	This url will receive the payment result from megapay (by redirecting)	Yes	String(255)
notiUrl	URL IPN receive the payment result from megapay (by http post)	Yes	String(255)
reqDomain	Merchant's website	Yes	String(255)
vat	VAT tax If there are no tax, vat = 0	No	Number(12)
fee	Transaction fee If there are no fee, fee = 0	Yes	Number(12)
notax	Merchant's tax number	No	Number(12)
description	Note information Without diacritical marks (Vietnamese character)	Yes	String(100)



merchantToken	<p>Merchant's signature: it was used to authenticate merchant's transaction.</p> <p>1. Single-message transaction (one step) :</p> <ul style="list-style-type: none"> - Formula for payment by card: Sha256(timeStamp + merTrxId + merId + amount + encodeKey) - Formula for payment by token: Sha256(timeStamp + merTrxId + merId + amount + payToken + encodeKey) payToken: encrypted by 3DES, key is 24 last characters of encodeKey <p>2. Dual-message transaction (two steps) :</p> <ul style="list-style-type: none"> - Formula for payment by card: Sha256(timeStamp + merTrxId + merId + amount + authorizations + encodeKey) - Formula for payment by token: Sha256(timeStamp + merTrxId + merId + amount + payToken + authorizations + encodeKey) payToken: encrypted by 3DES, key is 24 last characters of encodeKey 	Yes	String(255)
cardTypeToken	<p>Signature : Apply if Merchant using cardTypeValue Generated by this formula: Sha256(cardType + merTrxID + encodeKey)</p>	No	String(255)
reqServerIP	Merchant's server IP	No	String(15)
reqClientVer	Required version client	No	String(50)
userIP	User's IP	Yes	String(15)
userSessionID	User's ID Session	No	String(100)
userAgent	User's agent	No	String (100)
userLanguage	User's language. For default: "VN" VN = Vietnamese, EN = English, KO = Korean	Yes	String(2)
timeStamp	timeStamp Format: yyyyMMddHHmmss	Yes	String(13)
domesticToken	Not to be used	No	String(100)
payOption	<ul style="list-style-type: none"> - Payment and create token: PAY_CREATE_TOKEN - Payment with token: PAY_WITH_TOKEN Used when paying with payType = IC or DC 	No	String(100)
payToken	<p>This one was used when make a payment with tokenization</p> <p>Encrypted by 3DES</p> <p>Encrypted Key: 24 last character of encodeKey</p> <p>It is mandatory when payOption is PAY WITH TOKEN</p>	No	String(100)



userId	Used for tokenization Customer's code in merchant's system Required with: payOption = “PAY_CREATE_TOKEN”, “PAY_WITH_TOKEN”	No	String(40)
instmntType	Not used	No	String(2)
instmntMon	Not used	No	String(2)
merTrxId	Unique transaction ID generated by merchant (This trans ID will be used for inquiry transaction function) Format: merId + unique number	Yes	String(50)
windowColor	The color of the payment page (refer to the color table in the link below: https://html-color-codes.info/) Default “#cf5459”	Yes	String(7)
windowType	Merchant's device type 0 : PC, 1: Mobile	Yes	String(2)
vaStartDt	Effective date of deposit code (YYYYMMDDHHMMSS) Sample: 20190101000000 <ul style="list-style-type: none">- Pay by credit card (IC), installment (IS), ewallet(EW) and ATM card (DC), VNPayQR (QR): Optional- Pay by Deposit code (VA): Required	No	String(14)
vaEndDt	Expiration date of deposit code (YYYYMMDDHHMMSS) Sample: 20190101000000 <ul style="list-style-type: none">- Pay by credit card (IC), installment (IS), ewallet(EW) and ATM card (DC), VNPayQR (QR): Optional- Pay by Deposit code (VA): Required	No	String(14)
vaCondition	Conditions to check amount when receiving payment by Deposit code Default: 03 – transfer to the deposit code is accepted only when transferring amount is equal to the amount sent to MGP. <ul style="list-style-type: none">- Pay by credit card (IC), installment (IS), ewallet (EW) and ATM card (DC), VNPayQR (QR): Optional- Pay by Deposit code (VA): Required	No	String(2)
mer_temp01	Temporary field with additional requirements of Merchant	No	String
mer_temp02	Temporary field with additional requirements of Merchant	No	String
bankCode	Bank code send from Merchant. Megapay will automatically select bank by bankCode Applies to the following methods: - Apply for ATM card payment (DC): Only necessary when ATM bank is selected from merchant page (not recommended)	No	String(10)

	<p>The list of banks is in the appendix</p> <p>- Apply for installment payment (IS): Only necessary when installment bank information is shown on merchant page (when transmitting this field, it is required to transmit the termIs field).</p> <p>The list of banks is in the appendix</p> <p>- Apply for e-wallet payment (EW): Only necessary when merchants want to show specific EW provider although using more than 1 provider</p> <ul style="list-style-type: none"> + 'MOMO': pay with Momo wallet + 'ZALO': pay with ZaloPay wallet + 'MOCA': pay with MOCA wallet + 'SHPP': pay with ShopeePay + 'VTTP': pay with ViettelPay 		
subappid	Required only for ZaloPay and merchant category is "game provider"	No	String(100)
termIs	<p>Installment terms, example: 3, 6, 12 month</p> <p>Note: Only applicable for installment payment (IS) when merchant wants to display installment information on merchant page</p>	No	String(2)
subMerId	<p>Sub merchant ID</p> <p>Apply to merchants with sub-merchants</p>	No	String
site	<p>- Website name for making payments from</p> <p>- Can be blank</p> <p>Note: Only applicable to merchants with more than 1 entities to receive settlement from Epay.</p>	No	String(50)
soldToCode	<p>- Code to specify entities to receive settlements</p> <p>- Can be blank</p> <p>Note: Only applicable to merchants with more than 1 entities to receive settlement from Epay.</p>	No	String(20)
productList	<p>- List of products with related with data</p> <p>- Formatted as json Array</p> <p>- Mandatory for merchants with register more than 1 entities to receive settlements</p> <p>Note: Only applicable to merchants with more than 1 entities to receive settlement from Epay.</p>	No	String
hash	<p>- Formula: Sha256(merTrxId + site + soldToCode + productList + encodeKey)</p> <p>- Mandatory for merchants with register >= 2 Company receive settlement</p> <p>Note: Only applicable to merchants with more than 1 entities to receive settlement from Epay.</p>	No	String
authorizations	<p>Temporarily hold the amount to be paid on an international card, must take additional step to complete the payment (if you choose this method, you need to call Capture API to confirm the payment, refer to flow details in section 14):</p> <ul style="list-style-type: none"> + 1: Authorize transaction only; before sending separated capture request, transaction will not be completed. (Use 2-step payment with international cards) + Other: Transaction is completed via this request only; separated capture request is not needed. <p>Note: Only applicable for dual-message transactions (IC, IS)</p>	No	String(1)



The **productList** field is a json array **Only applicable to merchants with more than 1 entities to receive settlement from Epay**. Each element is a product object containing the following information:

Field name	Description	Required	Datatype
productCategory	- Product group name - Eg: Phone, Tablet Laptop	Yes	String(50)
companyCode	- This code is used to map with a number of account receiver money - Used for merchants with multiple accounts receiving payments, E.g: COMPANY_A or COMPANY_B Note: This code must be registered in MGP system before used	Yes	String(10)
productAmount	Sum amount of each product	Yes	String (12)
productCount	Quantity of each product	Yes	String (3)

json array template of **productList**:

```
[{"productCategory": "Phone", "companyCode": "COMPANY_A", "productAmount": "10000", "productCount": "1"}, {"productCategory": "Tablet", "companyCode": "COMPANY_B", "productAmount": "20000", "productCount": "2"}, {"productCategory": "Laptop", "companyCode": "COMPANY_B", "productAmount": "30000", "productCount": "3"}]
```

Sample code to create **productList** string with json Array format

```
var productList = [];
var product1 = {};
product1["productCategory"] = "Phone";
product1["companyCode"] = "COMPANY_A";
product1["productAmount"] = "10000";
product1["productCount"] = "1";
productList.push(product1);

var product2 = {};
product2["productCategory"] = "Tablet";
product2["companyCode"] = "COMPANY_B";
product2["productAmount"] = "20000";
product2["productCount"] = "2";
productList.push(product2);

var product3 = {};
product3["productCategory"] = "Laptop";
product3["companyCode"] = "COMPANY_B";
product3["productAmount"] = "30000";
product3["productCount"] = "3";
productList.push(product3);

console.log(JSON.stringify(productList));
```

5.2 Response



Special note 1: In case of payment with Deposit code, there are two types of successful result:



00_005: Deposit code has been issued and end user has NOT transferred the amount yet. Services or goods should NOT be provided to customers. 00_000: Deposit code has been issued and end user has transferred the amount already. In addition, when customers deposit the money into Deposit code, MegaPay will return the result to merchant via IPN.

Special note 2: Please make sure that requested amount and returned amount are equal. If not, consider the transaction as failed (Service or good should NOT be provided to customers) and consult with MegaPay team immediately.

After completing the payment steps, MegaPay will return the result to merchant by 2 ways simultaneously:

5.2.1 Redirect

Megapay will redirect the result to the url which was sent by merchant in the parameter named callBackUrl. No matter transaction is successful or failed, Megapay is always redirect to this call back page.

URL string for redirecting has format as below:

```
http://merchant_website/sampleMerchant.do?trxId=EPAY000001IC20191003103638179  
7&merId=EPAY000001&merTrxId=EPAY00000120191003054607&resultCd=00_000  
&resultMsg=SUCCESS&invoiceNo=OrdNo20191003054607&amount=100000&curren  
cy=VND&goodsNm=&payType=IC&domesticToken=&merchantToken=b5fd198a812a  
7596aaffff132d3b945ba34d7542c157460e4439bc29d6bcc77aa&transDt=20191003&trans  
Tm=103638&buyerFirstNm=bùi&buyerLastNm=luận&timeStamp=20191003054607&in  
stmntType=1&instmntMon=3&vaNumber=&vaName=&bankId=&bankName=&cardNo  
=xxxxxxxxxxxx0002&startDt=&endDt=&payToken=90665b34586d2a2d59cc1fbce3395  
c7f8c156991bd4dd172&userId=abc&subappid=1234&payOption=PAY_CREATE_TOK  
EN&userFee=0&userFeeIs=&termIs=&firstAmount=&nextAmount=&cardEnroll3DS=1
```

List of parameters:

Field name	Description	Required	Data type
trxId	Transaction code generated by MegaPay	Yes	String(30)
merId	Merchant ID on MegaPay system	Yes	String(10)
merTrxId	Transaction ID generated by Merchant	Yes	String(50)
resultCd	Transaction's result (refer to error code table)	Yes	String(6)

invoiceNo	Order code	Yes	String(40)
amount	Amount of payment, It is the amount of the merchant send request	Yes	String(12)
currency	Currency of each country	Yes	String(3)
goodsNm	Good's name	No	String(200)
payType	Payment method information - DC: Pay by ATM card - IC: Pay by credit card (Visa/master/JCB/...) - EW: Ewallet - IS: Installment - VA: Pay by Deposit code - QR: VNPAYQR	Yes	String(2)
merchantToken	Merchant's signature: it was used to authenticate merchant's transaction. 1. Single-message transaction (one step) a. If end-user has not to bear the payment fee (<i>condition: userFee is null or userFee = 0</i>) - Formula for payment by card: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{encodeKey})$ - Formula for payment by token: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{payToken} + \text{encodeKey})$ b. If end-user has to bear the payment fee (<i>condition: userFee > 0</i>) - Formula for payment by card: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{userFee} + \text{encodeKey})$ - Formula for payment by token: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{payToken} + \text{userFee} + \text{encodeKey})$ 2. Dual-message transaction (two steps) a. If end-user has not to bear the payment fee (<i>condition: userFee is null or userFee = 0</i>) - Formula for payment by card: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{authorizations} + \text{encodeKey})$ - Formula for payment by token: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{payToken} + \text{authorizations} + \text{encodeKey})$ b. If end-user has to bear the payment fee (<i>condition: userFee > 0</i>)	Yes	String(255)  In case of failed merchant token verification (mismatched or exception), let's call inquiry function again to get the exact status of transaction

	<p>- Formula for payment by card:</p> <p>Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + userFee + authorizations + encodeKey)</p> <p>- Formula for payment by token:</p> <p>Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + payToken + userFee + authorizations + encodeKey)</p> <p>Note: payToken: encrypted by 3DES</p>		
transDt	Transaction date	Yes	String(8)
transTm	Transaction time	Yes	String(6)
buyerFirstNm	Buyer's first name	No	String(30)
buyerLastNm	Buyer's last name	No	String(30)
timeStamp	timeStamp	Yes	String(13)
instmntType	Don't use	No	String(2)
instmntMon	Don't use	No	String(2)
cardNo	Card no Sample: ATM: 970400xxxxxx0018 Visa/master/JCB: xxxxxxxxxxxx0002	No	String(20)
bankId	<p>Bank code</p> <p>- With installment payment (IS) is the payment bank of the end user selected</p> <p>- With e-wallet payment (EW):</p> <ul style="list-style-type: none"> + 'MOMO': Pay with Momo wallet + 'ZALO': payment for ZaloPay wallet + 'MOCA': payment for MOCA wallet + 'SHPP': payment of ShopeePay + 'VTTP': payment of ViettelPay 	No	String(10)
bankName	<p>Bank name</p> <p>- With installment payment (IS) is the payment bank of the end user selected</p>	No	String(200)
vaNumber	Deposit code number: Returns when user chooses payment method by Deposit code (payType = VA)	No	String(30)
vaName	Deposit code name: Returns when user chooses payment method by Deposit code (payType = VA)	No	String(200)
mer_temp01	Temporary field with additional requirements of Merchant 1	No	String
mer_temp02	Temporary field with additional requirements of Merchant 2	No	String
startDt	V.A effective date (Returns when user chooses payment method by Deposit code (payType = VA))	No	String(14)
endDt	V.A expire date (Returns when user chooses payment method by Deposit code (payType = VA))	No	String(14)
payToken	It is returned when making a payment with	No	String(100)

	<p>tokenization</p> <p>Encrypted by 3DES</p> <p>Decrypted key: 24 first characters of encodeKey</p> <p>It is returned if payOption is PAY_CREATE_TOKEN or PAY_WITH_TOKEN</p>		
userId	<p>Used for tokenization</p> <p>Customer's code in merchant's system</p> <p>Required with: payOption = "PAY_CREATE_TOKEN", "PAY WITH TOKEN"</p>	No	String(40)
issueBankName	<ul style="list-style-type: none"> - Bank's name when paying by international card; - Return only when international card payment is successful 	No	String(1024)
cardType	<p>Card type:</p> <p>001: visa; 002: Master; 007: JCB</p>	No	String(3)
subappid	Subappid of zaloPay for merchant game	No	String
payOption	<ul style="list-style-type: none"> - Payment and create token: PAY_CREATE_TOKEN - Payment with token: PAY_WITH_TOKEN <p>Note:</p> <p>In case the Merchant pays with PAY_WITH_TOKEN when the token in Megapay has been deleted (Reason 90 days of no transactions and the Card expires), Megapay will display a card input form for the end-user:</p> <p></p> <ul style="list-style-type: none"> - If the end-user enters the card number that matches the old card number, Megapay will keep the token. - If the card number is different from the old card number, Megapay will generate a new token, map with the new card number, then MGP returns PAY_CREATE_TOKEN and returns the new payToken 	No	String(100)
userFee	<ul style="list-style-type: none"> - Fee payment of Enduser - Applies to Merchants has used payment share fee with Enduser 	No	String(12)
userFeeIs	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - Installment conversion fee that endser must bear 	No	String(12)
termIs	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - Term of repayment, such as 3, 6, 12 	No	String(2)
firstAmount	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - The amount of the first month enduser must pay 	No	String(12)



nextAmount	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - The amount of the months after the end user must pay 	No	String(12)
cardEnroll3DS	<ul style="list-style-type: none"> - Only apply for pay type is IC or IS - State of transaction, decide that transaction use 3DS or not <ul style="list-style-type: none"> + 0: Card does not enroll 3DS + 1: Card is enrolling 3DS, and merchant apply rule: "Ask card holder input 3DS password/OTP for all transaction" + 2: Card is enrolling 3DS, and merchant apply rule: "Card holder dose not need input 3DS password/OTP for all transaction" 	No	String(1)
status	<p>Transaction status:</p> <p>0: Paid 30: Waiting for review, encountering this condition need to wait for the final result from IPN</p>	Yes	String(2)
icIssue	<ul style="list-style-type: none"> - International cards issued in Vietnam or overseas: <ul style="list-style-type: none"> + IN : Cards issued in Vietnam + OUT : Cards issued overseas - Only applies to international cards (IC) 	No	String(3)
authorizations	<p>Temporarily hold the amount to be paid on an international card, must take additional step to complete the payment (if you choose this method, you need to call Capture API to confirm the payment, refer to flow details in section 14):</p> <p>+ 1: Authorize transaction only; before sending separated capture request, transaction will not be completed. (Use 2-step payment with international cards)</p> <p>+ Other: Transaction is completed via this request only; separated capture request is not needed.</p> <p>Note: Only applicable for dual-message transactions (IC, IS)</p>	No	String(1)

5.2.2 Send notification to Merchant via API (IPN - Instant Payment Notification)

When the payment transaction is successful, Megapay will return the result to merchant by Call Back link and Notify link simultaneously, these links will be sent to Megapay by Merchant in the payment request (Parameters named callBackUrl and notiUrl)

Method: POST

Data type: Json/application

Field name	Description	Required	Data type
resultCd	Transaction's status (According to the error code table)	Yes	String(6)
resultMsg	Status description	Yes	String(512)
merId	Merchant ID	Yes	String(10)
trxId	Transaction code generated by MegaPay	Yes	String(30)
invoiceNo	Order code generated by Merchant		
amount	Amount of payment, It is the amount of the merchant send request	Yes	String(12)
bankId	Bank code	Yes	String(4)

	<ul style="list-style-type: none"> - With installment payment (IS) is the payment bank of the end user selected - With e-wallet payment (EW): <ul style="list-style-type: none"> +) 'MOMO': Pay with Momo wallet +) 'ZALO': payment for ZaloPay wallet +) 'MOCA': payment for MOCA wallet +) 'SHPP': payment of ShopeePay +) 'VTTP': payment of ViettelPay 		
status	<p>Transaction status:</p> <p>0: Paid 2: Refund 30: Waiting for review, encountering this condition need to wait for the final result from IPN 31: Review result: Approved (transaction recorded successfully) 32: Review result: Rejection (transaction recorded as a failure and auto refund)</p>	Yes	String(1)
cardNo	Card no Sample: ATM: 970400xxxxxx0018 Visa/master/JCB: xxxxxxxxxxxx0002	Yes	String(20)
currency	Currency	Yes	String(3)
instmntMon	Don't use	No	String(2)
instmntType	Don't use	No	String(2)
goodsNm	Good's name	Yes	String(200)
buyerFirstNm	Buyer's first name	Yes	String(30)
buyerLastNm	Buyer's last name	Yes	String(30)
payType	<p>Payment method information</p> <ul style="list-style-type: none"> - DC: Pay by ATM card - IC: Pay by credit card (Visa/master/JCB,...) - EW: Ewallet - VA: Pay by Deposit code - IS: Installment - QR: VNPAYQR 	Yes	String(2)
trxDt	Transaction date (yyyymmdd)	Yes	String(8)
trxTm	Transaction time	Yes	String(6)
timeStamp	Time stamp		
merTrxId	Transaction code sent by Merchant	Yes	String(50)
merchantToken	<p>Merchant's signature: it was used to authenticate merchant's transaction.</p> <p>1. Single-message transaction (one step)</p> <p>a. If end-user has not to bear the payment fee (<i>condition: userFee is null or userFee = 0</i>)</p> <ul style="list-style-type: none"> - Formula for payment by card: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{encodeKey})$ - Formula for payment by token: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{payToken} + \text{encodeKey})$ <p>b. If end-user has to bear the payment fee (<i>condition: userFee > 0</i>)</p> <ul style="list-style-type: none"> - Formula for payment by card: $\text{Sha256}(\text{resultCd} + \text{timeStamp} + \text{merTrxId} + \text{trxId} + \text{merId} + \text{amount} + \text{userFee} + \text{encodeKey})$ 	 Yes	String(255) In case of failed merchant token verification (mismatched or exception), let's call inquiry function again to get the exact status of transaction



	<p>merId + amount + userFee + encodeKey)</p> <p>- Formula for payment by token: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + payToken + userFee + encodeKey)</p> <p>2. Dual-message transaction (two steps)</p> <p>a. If end-user has not to bear the payment fee (condition: userFee is null or userFee = 0)</p> <p>- Formula for payment by card: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + authorizations + encodeKey)</p> <p>- Formula for payment by token: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + payToken + authorizations + encodeKey)</p> <p>b. If end-user has to bear the payment fee (condition: userFee > 0)</p> <p>- Formula for payment by card: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + userFee + authorizations + encodeKey)</p> <p>- Formula for payment by token: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + payToken + userFee + authorizations + encodeKey)</p> <p>Note: payToken: encrypted by 3DES</p>		
startDt	Effective date V.A (Returns when user chooses payment method by Deposit code (payType = VA))	No	String(14)
endDt	Expire date V.A (Returns when user chooses payment method by Deposit code (payType = VA))	No	String(14)
vaName	Deposit code name (Returns when user chooses payment method by Deposit code (payType = VA))	No	String(50)
vaNumber	Deposit code number	No	String(30)
payToken	<p>It is returned when making a payment with tokenization</p> <p>Encrypted by 3DES</p> <p>Decrypted key: 24 first characters of encodeKey</p> <p>It is returned if payOption is PAY_CREATE_TOKEN or PAY_WITH_TOKEN</p>	No	String(100)
userId	<p>Used for tokenization</p> <p>Customer's code in merchant's system</p> <p>Required with: payOption = "PAY_CREATE_TOKEN", "PAY_WITH_TOKEN"</p>	No	String(40)
issueBankName	<ul style="list-style-type: none"> - Bank's name when paying by international card; - Return only when international card payment is successful 	No	String(1024)
cardType	Card type: 001: visa; 002: Master; 007: JCB	No	String(3)
subappid	Subappid of zaloPay for merchant game	No	String
userFee	<ul style="list-style-type: none"> - Fee payment of Enduser - Applies to Merchants has used payment share fee with Enduser 	No	String(12)



userFeeIs	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - Installment conversion fee that enduser must bear 	No	String(12)
termIs	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - Term of repayment, such as 3, 6, 12 	No	String(2)
firstAmount	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - The amount of the first month enduser must pay 	No	String(12)
nextAmount	<ul style="list-style-type: none"> - Only applicable to installment payments (IS) - The amount of the months after the end user must pay 	No	String(12)
payOption	<p> - Payment and create token: PAY_CREATE_TOKEN - Payment with token: PAY_WITH_TOKEN </p> <p>Note: In case the Merchant pays with PAY_WITH_TOKEN when the token in Megapay has been deleted (Reason 90 days of no transactions and the Card expires), Megapay will display a card input form for the end-user:  - If the end-user enters the card number that matches the old card number, Megapay will keep the token. - If the card number is different from the old card number, Megapay will generate a new token, map with the new card number, then MGP returns PAY_CREATE_TOKEN and returns the new payToken </p>	No	String(100)
cardEnroll3DS	<ul style="list-style-type: none"> - Only apply for pay type is IC or IS - State of transaction, decide that transaction use 3DS or not <ul style="list-style-type: none"> +) 0: Card does not enroll 3DS +) 1: Card is enrolling 3DS, and merchant apply rule: "Ask card holder input 3DS password/OTP for all transaction" +) 2: Card is enrolling 3DS, and merchant apply rule: "Card holder dose not need input 3DS password/OTP for all transaction" 	No	String(1)
icIssue	<ul style="list-style-type: none"> - International cards issued in Vietnam or overseas: <ul style="list-style-type: none"> + IN : Cards issued in Vietnam + OUT : Cards issued overseas - Only applies to international cards (IC) 	No	String(3)
authorizations	<p>Temporarily hold the amount to be paid on an international card, must take additional step to complete the payment (if you choose this method, you need to call Capture API to confirm the payment, refer to flow details in section 14):</p> <p> +) 1: Authorize transaction only; before sending separated capture request, transaction will not be completed. (Use 2-step payment with international cards) +) Other: Transaction is completed via this request only; separated capture request is not needed. </p> <p>Note: Only applicable for dual-message transactions (IC, IS)</p>	No	String(1)



Note: Merchant call inquiry transaction function again to check the transaction status before providing service to customer

Json string template:

```
{
  "vaName": "null",
  "accessUrl": "http://172.16.10.132:5858/NotiUrl",
  "buyerLastNm": "luzn",
  "trxTm": "103638",
  "cardNo": "xxxxxxxxxxxx0002",
  "vaNumber": "null",
  "matchStatus": "1",
  "payType": "IC",
  "currency": "VND",
  "instmntMon": "3",
  "invoiceNo": "OrdNo20191003054607",
  "merchantToken": "b5fd198a812a7596aafff132d3b945ba34d7542c157460e4439bc29d6bcc77aa",
  "amount": "100000",
  "goodsNm": "null",
  "resultCd": "00_000",
  "merTrxId": "EPAY00000120191003054607",
  "trxId": "EPAY000001IC201910031036381797",
  "userId": "hnluanbvb",
  "instmntType": "1",
  "resultMsg": "SUCCESS",
  "timeStamp": "20191003054607",
  "trxDt": "20191003",
  "bankId": "STBM",
  "merId": "EPAY000001",
  "buyerFirstNm": "b?i",
  "domesticToken": "null",
  "targetUrl": "http://172.16.10.132:5858/NotiUrl",
  "status": "0",
  "payToken": "90665b34586d2a2d59cc1fbce3395c7f8c156991bd4dd172",
  "subappid": "1234",
  "userFee": "0",
  "userFeeIs": "0",
  "termIs": "6",
  "firstAmount": "20000",
  "nextAmount": "16000",
  "payOption": "PAY_CREATE_TOKEN",
  "cardEnroll3DS": "1"
}
```

6 Pay with Tokenization (Direct API Payment)



Only applicable for merchants using recurring payment. This function does not work if 3DS is turn on.

URL https://sandbox.megapay.vn/pg_was/payWithTokenAPI.do

Method: POST

6.1 Request

Data type: Json/application

Tên trường	Mô tả	Bắt buộc	Kiểu dữ liệu
merId	Unique ID for Merchant Sample: [String] "MEGAPAYTES"	Yes	String(10)
merTrxId	Unique transaction ID generated by merchant (This trans ID will be used for inquiry transaction function) Format: merId + unique number	Yes	String(50)
amount	Amount of money for payment (good's price) Sample: [String] "500000"	Yes	String(12)
currency	Currency of each country Only allow "VND" now	Yes	String(3) - Fixed
payType	Payment method: this function only support IC : Credit card (Visa/master/JCB...)	Yes	String(2)
timeStamp	timeStamp	Yes	String(13)
fee	Fee	No	String(12)



vat	VAT	No	String(12)
notax	Tax number	No	String(12)
invoiceNo	Invoice number	Yes	String (40)
goodsNm	Good's name	Yes	String(200)
notiUrl	This url will receive the payment result from megapay (by redirecting)	Yes	String(255)
merchantToken	<p>Merchant's signature 1. Single-message transaction (one step) Sha256(timeStamp + merTrxId + merId + amount +payToken+ encodeKey)</p> <p>2. Dual-message transaction (two steps) Sha256(timeStamp + merTrxId + merId + amount +payToken+ authorizations + encodeKey)</p>	Yes	String(255)
payOption	<p>payOption = “PAY_WITH_TOKEN_API”: If it’s a payment with token transaction</p> <p><i>Now, tokenization is only being supported by international payment method</i></p>	Yes	String(100)
payToken	<p>This one was used when make a payment with tokenization</p> <p><i>It's required if payOptio = “PAY_WITH_TOKEN_API” (Only apply for international payment now)</i></p> <p>Encrypted by 3DES Encrypted Key: 24 last character of encodeKey</p>	Yes	String(100)
userId	<p>Used for tokenization</p> <p>Customer’s code in merchant’s system</p> <p><i>It's required if payOption = “PAY WITH TOKEN API”</i></p>	Yes	String(40)
authorizations	<p>Temporarily hold the amount to be paid on an international card, must take additional step to complete the payment (if you choose this method, you need to call Capture API to confirm the payment, refer to flow details in section 14):</p> <ul style="list-style-type: none"> +) 1: Authorize transaction only; before sending separated capture request, transaction will not be completed. (Use 2-step payment with international cards) +) Other: Transaction is completed via this request only; separated capture request is not needed. <p>Only apply with (IC, IS)</p>	No	String(1)

6.2 Response

Data type: Json/application

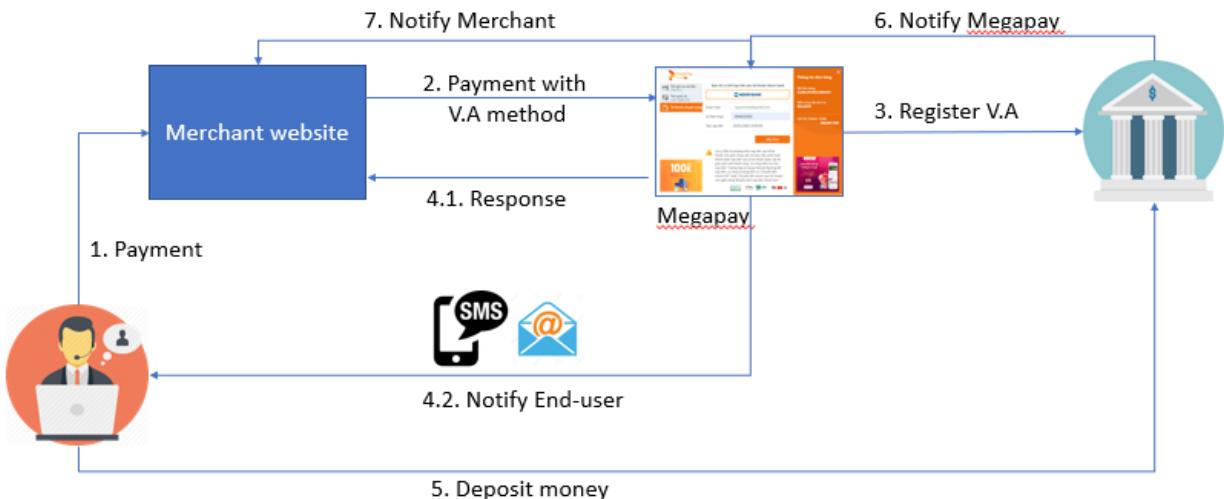
Tên trường	Mô tả	Bắt buộc	Kiểu dữ liệu
resultCd	Transaction’s status (According to the error code table)	Yes	String(6)
resultMsg	Description	Yes	String(512)
merId	Merchant ID	Yes	String(10)
trxId	Transaction code generated by MegaPay	Yes	String(30)
invoiceNo	Invoice number	Yes	String (40)
amount	Amount of payment	Yes	String(12)
bankId	Bank ID	No	String(4)
status	Transaction status:	Yes	String(2)

	0: Paid 2: Refund 30: Waiting for review, encountering this condition need to wait for the final result from IPN		
cardNo	Card number Sample: ATM: 970400xxxxxx0018 Visa/master/JCB: xxxxxxxxxxxx0002	Yes	String(20)
currency	currency	Yes	String(3)
instmntMon	Not use	No	String(2)
instmntType	Not use	No	String(2)
goodsNm	Good's name	Yes	String(200)
buyerFirstNm	Buyer's first name	Yes	String(30)
buyerLastNm	Buyer's last name	Yes	String(30)
payType	Payment method information: IC: Pay by credit card (Visa/master/JCB,...)	Yes	String(2)
transDt	Transaction date	Yes	String(8)
transTm	Transaction time	Yes	String(6)
timeStamp	Time stamp		
merTrxId	Transaction ID generated by Merchant	Yes	String(50)
merchantToken	<p>1. Single-message transaction (one step)</p> <p>a. If end-user has not to bear the payment fee (condition: userFee is null or userFee = 0) Sha256(resultCd +timeStamp +merTrxId +trxId +merId + amount + payToken + encodeKey)</p> <p>b. If end-user has to bear the payment fee (condition: userFee > 0) Sha256(resultCd +timeStamp +merTrxId +trxId +merId + amount + payToken + userFee + encodeKey)</p> <p>2. Dual-message transaction (two steps)</p> <p>a. If end-user has not to bear the payment fee (condition: userFee is null or userFee = 0) Sha256(resultCd +timeStamp +merTrxId +trxId +merId + amount + payToken + authorizations + encodeKey)</p> <p>b. If end-user has to bear the payment fee (condition: userFee > 0) Sha256(resultCd +timeStamp +merTrxId +trxId +merId + amount + payToken + userFee + authorizations + encodeKey)</p> <p>Note payToken: Encrypted by 3DES</p>	Yes	String(255)  In case of failed merchant token verification (mismatched or exception), let's call inquiry function again to get the exact status of transaction
payToken	<i>This one will be returned when</i> payOption = "PAY_WITH_TOKEN_API" Encrypted by 3DES Decrypted key: 24 first characters of encodeKey	No	String(100)
userId	<i>This one will be returned when</i> payOption = "PAY WITH TOKEN API"	No	String(40)
userFee	- Fee payment of Enduser - Applies to Merchants has used payment share fee with Enduser	No	String(12)



cardEnroll3DS	<ul style="list-style-type: none">- Only apply for pay type is IC or IS- State of transaction, decide that transaction use 3DS or not<ul style="list-style-type: none">+) 0: Card does not enroll 3DS+) 1: Card is enrolling 3DS, and merchant apply rule: "Ask card holder input 3DS password/OTP for all transaction"+) 2: Card is enrolling 3DS, and merchant apply rule: "Card holder dose not need input 3DS password/OTP for all transaction"	No	String(1)
authorizations	<p>Temporarily hold the amount to be paid on an international card, must take additional step to complete the payment (if you choose this method, you need to call Capture API to confirm the payment, refer to flow details in section 14):</p> <ul style="list-style-type: none">+) 1: Authorize transaction only; before sending separated capture request, transaction will not be completed. (Use 2-step payment with international cards)+) Other: Transaction is completed via this request only; separated capture request is not needed. <p>Note: Only applicable for dual-message transactions (IC, IS)</p>	No	String(1)

7 Pay by Deposit code:



Step	Description
1	Customer selects goods which are need to be paid on the Merchant's page and click "Payment"
2	Customer chooses payment method named "Deposit code" (pay by Deposit code)
3	Epay connects to bank to assign a Deposit code for an order code
4.1	Megapay return the result of payment made by Deposit code to merchant by redirecting to merchant's page, merchant displays the result of payment for end-user (Main information: Deposit code, Account name, Amount of money End-user must transfer, order code)
	<p>Notes: In this step, Megapay just assign a Deposit code for the order, the returned result is "Successful" it's mean assigned successful, it does NOT mean the order has paid already, Merchants should NOT provide goods or service to end-user in this step</p>
4.2	Megapay also sends Email to End-user (Main info: Deposit code number, account name, amount of money end-user, order code) <i>Sending Email can be either enable or disable depend on merchant's demand</i>
5	After receiving Deposit code number, end-user transfers the amount to the Deposit code (by the following channels: bank counter, internet banking, ATM, etc.)
6	When a transfer is made, Bank will notify for Megapay
7	MegaPay get the order information base on the assigned information between order code – Deposit code number. Megapay notify for merchant via API notify (IPN – described in step 4.2.2) And now, Merchant provides goods – service to end-user



Note: Deposit code number is mapped with an order; it's useable for one time only



8 Refund transaction



This function doesn't apply for Deposit code

URL: https://sandbox.megapay.vn/pg_was/cancel/paymentCancel.do

Method: POST

8.1 Request

Field name	Description	Required	Data type
trxId	Unique ID for transaction, original transaction code	Yes	String(30)
merId	Unique ID for Merchant	Yes	String(10)
merTrxId	Transaction code generated by merchant	Yes	String(50)
amount	- Sum amount of money to refund. Amount of refund must be less than or equal to original transaction amount. - Merchant is allowed to refund multiple times so that total amount of refund is not greater than original transaction amount.	Yes	String(12)
payType	Payment method: - IC : Pay by Credit Card (Visa/master/JCB,...) - DC : Pay by ATM - EW: Pay by Ewallet - IS: Pay by Insallment - QR: Pay by VNPAYQR	Yes	String(2)
cancelMsg	The reason of canceling transaction	Yes	String(100)
timeStamp	timeStamp	Yes	String(13)
fee	Canceling transaction fee	No	String(12)
vat	Vat tax	No	String(12)
notax	No Tax Amount	No	String(12)
cancelServiceIp	IP of canceling server	No	String(15)
cancelUserId	ID of canceling user	No	String(30)
cancelUserIp	IP of canceling user	No	String(15)
cancelUserInfo	Information of canceling user	No	String(100)
merchantToken	Sha256(timeStamp + merTrxId + trxId + merId + amount + encodeKey)	Yes	String(255)
cancelPw	Merchant's cancelling password via API (Url encoded)	Yes	String(255)
refundData	- List of partial refund amounts with corresponding companyCode - Formatted as json Array - Mandatory for merchants with more than 1 entities to receive settlement from Epay. Only applicable to merchants with more than 1 entities to receive settlement from Epay.	No	String
hash	- Formula: Sha256(merTrxId + refundData+ encodeKey) - Mandatory for merchants with register >= 2 Company receive settlement	No	String



	Note: Only applicable to merchants with more than 1 entities to receive settlement from Epay.		
--	---	--	--

The **refundData** field is a json array (**Only applicable to merchants with more than 1 entities to receive settlement from Epay**), each element is a company object containing the following information:

Field name	Description	Required	Datatype
companyCode	companyCodes of products for refund	Yes	String(10)
partialAmount	Sum of refund amount for each company code	Yes	String (12)

json array template of **refundData**:

```
[{"companyCode": "COMPANY_A", "partialAmount": "10000"}, {"companyCode": "COMPANY_B", "partialAmount": "50000"}]
```

Example:

If an original transaction amount is 120000 and contains 4 products,

- productA, COMPANY_A, 10000
- productB, COMPANY_A, 20000
- productC, COMPANY_B, 40000
- productD, COMPANY_B, 50000

a. you want to refund productB, “amount” would be 20000 and refundData would be

```
[{"companyCode": "COMPANY_A", "partialAmount": "20000"}]
```

b. you want to refund productA and productB, “amount” would be 30000 and refundData would be

```
[{"companyCode": "COMPANY_A", "partialAmount": "30000"}]
```

c. you want to refund productA and productC, “amount” would be 50000 and refundData would be

```
[{"companyCode": "COMPANY_A", "partialAmount": "10000"}, {"companyCode": "COMPANY_B", "partialAmount": "40000"}]
```

d. you want to refund productA, productC, and productD, “amount” would be 100000 and refundData would be

```
[{"companyCode": "COMPANY_A", "partialAmount": "10000"}, {"companyCode": "COMPANY_B", "partialAmount": "90000"}]
```

e. you want to refund all, “amount” would be 120000 and refundData would be

```
[{"companyCode": "COMPANY_A", "partialAmount": "30000"}, {"companyCode": "COMPANY_B", "partialAmount": "90000"}]
```



8.2 Response

If refund successful, Megapay also send an IPN to Merchant – list parameters are the same with IPN of Payment method. Merchant should be call Inquiry transaction function when receive IPN.

Field name	Description	Required	Data type
trxId	Code of canceled transaction	No	String(30)
invoiceNo	Order code	No	String(40)
cancelTrxId	Code of canceling transaction (Generated by megapay)	No	String(30)
resultCd	Result of canceling transaction 00_000: Success Others: fail	Yes	String(6)
resultMsg	Message result of canceling transaction	Yes	String(100)
trxDt	Transaction date yyyymmdd	No	String(8)
trxTm	Transaction time hh24miss	No	String(6)
timeStamp	Time stamp	No	String(13)
description	Description	No	String(100)
amount	Refunded amount after canceling transaction	No	String(12)
merTrxId	Transaction code generated by Merchant	No	String(50)
merchantToken	Sha256(resultCd +timeStamp +merTrxId +trxId +merId +amount +encodeKey)	No	String(255)  In case of failed merchant token verification (mismatched or exception), let's call inquiry function again to get the exact status of transaction
payMessage	CC_DCAN	Yes	String(7)

Message template

- **Successful refunding:**

```
{"trxId": "SITNAPAS01DC201908161421387007", "invoiceNo": "OrdNo20190816092739", "cancelTrxId": "SITNAPAS01DC201908161430527012", "resultCd": "00_000", "resultMsg": "SUCCESS", "trxDt": "20190816", "trxTm": "142138", "description": "fdsafsd", "amount": "2000", "merchantToken": "151b400317c6f82340891bc892e9b33c4c6cdb055689f9683444dc8a800fc56", "merTrxId": "MERTRXID20190816092739", "payMessage": "CC_DCAN", "timeStamp": "20190816092739"}
```

- **Failed refunding:**

```
{"trxId": "SITNAPAS01DC201908161421387007", "invoiceNo": "OrdNo20190816092739", "cancelTrxId": null, "resultCd": "CC_119", "resultMsg": "Amount The amount you"}
```



```
entered is larger than the amount you want to
cancel.", "trxDt": "20190816", "trxTm": "142138", "description": "fdsafsd", "amount"
:"500000", "merchantToken": "b4a8d1d78a80e139ebbf65196cf18c96a32609c6d39838f173
3a783858f82667", "merTrxId": "MERTRXID20190816092739", "payMessage": "CC_DCAN", "t
imeStamp": "20190816092739"}
```

- Failed refunding:

```
{"trxId": null, "invoiceNo": null, "cancelTrxId": null, "resultCd": "CC_110", "result
Msg": "Invalid transaction
number.", "trxDt": null, "trxTm": null, "description": null, "amount": null, "merchant
Token": "9368f3783210175e5a457ac4c2cf0ba759cee9d3c61da8a5c38cbed33715ee48", "me
rTrxId": null, "payMessage": "CC_ICAN", "timeStamp": "1566294563371"}
```

9 Inquiry transaction status



If merchants don't receive the result of transactions (because of timeout, user interrupted connection, connection error), Merchants have to call this function to check the final status of transaction

9.1 Request

API URL: https://sandbox.megapay.vn/pg_was/order/trxStatus.do

Method = POST

List of parameter:

Field name	Description	Required	Data type
merId	Merchant code in MegaPay system	Yes	String(10)
merTrxId	Transaction code of merchant, used to inquiry transaction's information	Yes	String(50)
merchantToken	Check sum data: SHA256(timeStamp + merTrxId + merId + encodeKey)	Yes	String(255)
timeStamp	timeStamp	Yes	String(13)

Template of request:

merId=EPAYABC111&merTrxId=EPAYABC11132019164952H5zcZ0&merchantTok
en=d0c6e1ac9a18e2f9cb7b92115a0f2871fd415dc05fda0f94d0d341a68a6b039e&timeSta
mp=1551436017653

9.2 Response

List of parameters:

Field name	Description	Required	Data type
resultCd	Result code of inquiry transaction. If result code is "00_000" (inquiry transaction is successful), additional data string, which contains the information of inquired transaction, will be included.	Yes	String(6)
data	Information of inquired transaction		
trxId	Transaction code generated by MegaPay	Yes	String(30)
merId	Merchant ID	Yes	String(10)
currency	Currency type. Only support VND now	Yes	String(3)
amount	Amount of payment, it is the amount of the merchant send request	Yes	String(12)
invoiceNo	Order code	Yes	String(40)
goodsNm	Good's name	Yes	
payType	Payment method: IC : Credit Card (Visa/master/JCB...) IS: Installment DC : ATM card EW: Ewallet VA : Deposit code QR: VNPPAYQR	Yes	String(2)
buyerFirstNm	Buyer's first name	Yes	String(30)
buyerLastNm	Buyer's last name	Yes	String(30)
buyerPhone	Buyer's phone	No	String(15)
buyerEmail	Buyer's email	No	String(40)
resultCd	Result of payment transaction - 00_000: Success - 99: Pending - Others: Fail	Yes	String(6)
resultMsg	Description corresponding to resultCd	Yes	String(200)
status	Detailed status of the transaction +) -3: Original transaction is failed +) -2: Original transaction is pending +) -1: Original transaction not found, inquiry after 30 minutes to check the final status of transaction. (When inquired after 30 minutes, if it's still -1, Merchant should record the transaction as fail) +) 0: Original transaction is successful and not refunded yet +) 1: Original transaction is successful and Installment conversion is canceled or rejected +) 2: Original transaction is successful, and refund is done. +) 5: Deposit code is issued, but end user has not deposited money yet +) 30: Waiting for review, final result will be sent via IPN after approval is done on MMS +) 31: Review result is "approved" +) 32: Review result is "rejected"	Yes	String(2)

	Note: Pay attention to this field to see what step the last status of the transaction is in		
twoStepStatus	2-step transactions will have data in this field (NULL for 1 step transactions) +) 1: Authorized only +) 2: Captured (previously authorized) +) 3: Authorization is canceled +) 4: Capture is rejected	No	String(1)
bankId	Bank code	Yes	String(10)
cardNo	Card no	Yes	String(20)
trxDt	Transaction date	Yes	String(8)
trxTm	Transaction time	Yes	String(6)
cancelDt	Refund date (Only refunded transaction)	No	String(8)
cancelTm	Refund time (Only refunded transaction)	No	String(6)
payMessage	Default value is "TX_CHK"	Yes	String
merTrxId	Transaction code generated by Merchant	Yes	String(50)
vaNumber	V.A number (Only Deposit code transaction)	No	String(30)
startDt	Effective date V.A (Only Deposit code transaction)	No	String(14)
endDt	Expire date V.A (Only Deposit code transaction)	No	String(14)
vaName	Deposit code name (Only Deposit code transaction)	No	String(50)
timeStamp	Time stamp	No	String(13)
merchantToken	<p>Merchant's signature: it was used to authenticate merchant's transaction.</p> <p>a. The formula does not configuration the end user to bear the fee</p> <ul style="list-style-type: none"> - Formula for payment by card: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + encodeKey) - Formula for payment by token: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + payToken + encodeKey) <p>b. Formula with end user configuration bear the fee</p> <ul style="list-style-type: none"> - Formula for payment by card: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + userFee + encodeKey) - Formula for payment by token: Sha256(resultCd + timeStamp + merTrxId + trxId + merId + amount + payToken + userFee + encodeKey) <p>Note: payToken: encrypted by 3DES</p>	No	String(255)  In case of failed merchant token verification (mismatched or exception), please contact with Megapay
remainAmount	<p>Remain amount.</p> <p>In case of partial refund, this amount will be the remaining amount after partial refund</p> <p>In case of transaction haven't refunded yet, the remaining amount will equal the transaction amount</p>	Yes	String(12)
payToken	<p>This one was used when make a payment with tokenization</p> <p>Encrypted by 3DES</p> <p>Decrypted key: 24 first character of encodeKey</p>	No	String(100)

	Apply when paying with Token: PAY_WITH_TOKEN		
userId	Used for tokenization Customer's code in merchant's system Required with: payOption = " PAY_CREATE_TOKEN ", " PAY_WITH_TOKEN "	No	String(40)
userFee	- Fee payment of Enduser - Applies to Merchants has used payment share fee with Enduser	No	String(12)
userFeeIs	- Only applicable to installment payments (IS) - Installment conversion fee that enduser must bear	No	String(12)
termIs	- Only applicable to installment payments (IS) - Term of repayment, such as 3, 6, 12	No	String(2)
firstAmount	- Only applicable to installment payments (IS) - The amount of the first month enduser must pay	No	String(12)
nextAmount	- Only applicable to installment payments (IS) - The amount of the months after the end user must pay	No	String(12)
payOption	<ul style="list-style-type: none"> - Payment and create token: PAY_CREATE_TOKEN - Payment with token: PAY_WITH_TOKEN <p>Note: In case the Merchant pays with PAY_WITH_TOKEN when the token in Megapay has been deleted (Reason 90 days of no transactions and the Card expires), Megapay will display a card input form for the end-user:  <ul style="list-style-type: none"> - If the end-user enters the card number that matches the old card number, Megapay will keep the token. - If the card number is different from the old card number, Megapay will generate a new token, map with the new card number, then MGP returns PAY_CREATE_TOKEN and returns the new payToken </p>	No	String(100)
cardEnroll3DS	<ul style="list-style-type: none"> - Only apply for pay type is IC or IS - State of transaction, decide that transaction use 3DS or not <ul style="list-style-type: none"> + 0: Card does not enroll 3DS + 1: Card is enrolling 3DS, and merchant apply rule: "Ask card holder input 3DS password/OTP for all transaction" + 2: Card is enrolling 3DS, and merchant apply rule: "Card holder dose not need input 3DS password/OTP for all transaction" 	No	String(1)

For 2 step transactions (payType = IC or IS), merchants need to pay attention to both **status** and **twoStepStatus** to check the final status of the transaction.

status	twoStepStatus	Transaction status	Note
--------	---------------	--------------------	------



0	NULL	successful	for 1 step transaction, successful transaction
0	1	pending	for 2 step transaction, needs to send capture to complete the transaction
0	2	successful	for 2 step transaction, both auth & capture done
0	3	fail	for 2 step transaction, authorization is canceled
0	4	fail	for 2 step transaction, capture failed
31	NULL	successful	for 1 step transaction, approved after review, transaction successful
31	1	pending	for 2 step transaction, approved after review, needs to send capture to complete the transaction
31	2	successful	for 2 step transaction, approved after review, both auth & capture done
31	3	fail	for 2 step transaction, approved after review, authorization is canceled
31	4	fail	for 2 step transaction, approved after review, capture failed

Json string template

Inquiry success:	
Original transaction fails	<pre>{"resultCd": "00_000", "data": {"trxId": "-1", "merId": "EPAY000001", "resultCd": "PG_ER5", "resultMsg": "Customer cancellation", "status": "-3", "timeStamp": "1570075200112", "merchantToken": "01de67683045c8f8f3a4e87f436049f135f3f7cabf639d0d9784055388bdfd66", "payMessage": "TX_CHK", "merTrxId": "EPAY00000120191003054606"}}</pre> <p>Note: Empty data fields have been removed from the return string</p>
Original transaction is pending	<pre>{"resultCd": "00_000", "data": {"merchantToken": "c6a54ced9b628da494b8001cdf038ae9dd59c382b7beaf23483069117576b5db", "amount": "10000", "goodsNm": "Tea milk", "resultCd": "99", "buyerEmail": "FOABPNGE@gmail.com", "buyerLastNm": "Nguyen", "merTrxId": "EPAY000001_200914133415_funfv", "trxId": "EPAY000001DC202009141334421649", "resultMsg": "Giao d?ch ?ang ch? x? l?y", "timeStamp": "1600065260940", "payType": "DC", "merId": "EPAY00001", "currency": "VND", "invoiceNo": "Order_200914133415_25740", "buyerFirstNm": "Van A", "payMessage": "TX_CHK", "status": "-2"}}</pre> <p>Note: Empty data fields have been removed from the return string</p>
Transaction not found	<pre>{"resultCd": "00_000", "data": {"trxId": "-1", "merId": "EPAY000001", "resultCd": "OR_140", "resultMsg": "Transaction does not exist", "status": "-1", "timeStamp": "1570075200112", "merchantToken": "01de67683045c8f8f3a4e87f436049f135f3f7cabf639d0d9784055388bdfd66", "payMessage": "TX_CHK", "merTrxId": "EPAY00000120191003054606"}}</pre> <p>Note: Empty data fields have been removed from the return string</p>



Original transaction is successful (no refund)	<pre>{"resultCd": "00_000", "data": {"trxId": "EPAY000001IC201910031036381797", "merId": "EPAY000001", "currency": "VND", "amount": "100000", "invoiceNo": "OrdNo20191003054607", "payType": "IC", "buyerFirstNm": "bùi", "buyerLastNm": "luận", "buyerEmail": "builuan@gmail.com", "resultCd": "00_000", "resultMsg": "paid", "status": "0", "bankId": "STBM", "cardNo": "xxxxxxxxxxxx0002", "trxDt": "20191003", "trxTm": "103652", "timeStamp": "1570075152253", "merchantToken": "1b7ad56da9f079b4bc8bf85e42222476f84c90beb16bddea0048d449916b3fd0", "payMessage": "TX_CHK", "merTrxId": "EPAY00000120191003054607", "remainAmount": "0", "payToken": "90665b34586d2a2d59cc1fbce3395c7f8c156991bd4dd172", "userId": "hnluanbv", "userFee": "100", "payOption": "PAY_CREATE_TOKEN", "cardEnroll3DS": "1"}}</pre> <p>Note: Empty data fields have been removed from the return string</p>
Original transaction is successful (refund already made)	<pre>{"resultCd": "00_000", "data": {"trxId": "EPAY000001IC201910031036381797", "merId": "EPAY000001", "currency": "VND", "amount": "100000", "invoiceNo": "OrdNo20191003054607", "payType": "IC", "buyerFirstNm": "bùi", "buyerLastNm": "luận", "buyerEmail": "builuan@gmail.com", "resultCd": "00_000", "resultMsg": "refund", "status": "2", "bankId": "STBM", "cardNo": "xxxxxxxxxxxx0002", "trxDt": "20191003", "trxTm": "103652", "timeStamp": "1570075152253", "merchantToken": "1b7ad56da9f079b4bc8bf85e42222476f84c90beb16bddea0048d449916b3fd0", "cancelDt": "20191003", "cancelTm": "104904", "payMessage": "TX_CHK", "merTrxId": "EPAY00000120191003054607", "remainAmount": "0", "payToken": "90665b34586d2a2d59cc1fbce3395c7f8c156991bd4dd172", "userId": "hnluanbv", "payOption": "PAY_CREATE_TOKEN", "cardEnroll3DS": "1"}}</pre> <p>Note: Empty data fields have been removed from the return string</p>
Original transaction is successful (IS cancel/Reject already made)	<pre>{"resultCd": "00_000", "data": {"trxId": "EPAY000001IC201910031036381797", "merId": "EPAY000001", "currency": "VND", "amount": "1000000", "invoiceNo": "OrdNo20191003054607", "payType": "IS", "buyerFirstNm": "bùi", "buyerLastNm": "luận", "buyerEmail": "builuan@gmail.com", "resultCd": "00_000", "resultMsg": "IS cancel/Reject", "status": "1", "bankId": "STBM", "cardNo": "xxxxxxxxxxxx0002", "trxDt": "20191003", "trxTm": "103652", "timeStamp": "1570075152253", "merchantToken": "1b7ad56da9f079b4bc8bf85e42222476f84c90beb16bddea0048d449916b3fd0", "cancelDt": "20191003", "cancelTm": "104904", "payMessage": "TX_CHK", "merTrxId": "EPAY00000120191003054607", "remainAmount": "0", "payToken": "90665b34586d2a2d59cc1fbce3395c7f8c156991bd4dd172", "userId": "hnluanbv", "userFee": "0", "userFeeIs": "200000", "termIs": "9", "firstAmount": "3000000", "nextAmount": "1000000", "cardEnroll3DS": "1"}}</pre> <p>Note: Empty data fields have been removed from the return string</p>
Deposit code successfully registered, but not yet deposited by end-user	<pre>{"resultCd": "00_000", "data": {"trxId": "EPAY000001IC201910031036381797", "merId": "EPAY000001", "currency": "VND", "amount": "100000", "invoiceNo": "OrdNo20191003054607", "payType": "IC", "buyerFirstNm": "bùi", "buyerLastNm": "luận", "buyerEmail": "builuan@gmail.com", "resultCd": "00_000", "resultMsg": "pending", "status": "5", "bankId": "STBM", "cardNo": "xxxxxxxxxxxx0002", "trxDt": "20191003", "trxTm": "103652", "timeStamp": "1570075152253", "merchantToken": "1b7ad56da9f079b4bc8bf85e42222476f84c90beb16bddea0048d449916b3fd0", "payMessage": "TX_CHK", "merTrxId": "EPAY00000120191003054607"}}</pre>



	54607", "remainAmount": "0", "payToken": "90665b34586d2a2d59cc1fbce3395c7f8c156991bd4dd172", "userId": "hnluanbv" } }
Note: Empty data fields have been removed from the return string	
Inquiry fail	
{ "resultCd": "OR_137", "resultMsg": "Check Trx Order Error." }	

10 Installment cancel (IS cancel)



This function only applies for **Installment**

URL: https://sandbox.megapay.vn/pg_was/cancel/paymentISCancel.do

Method: POST

10.1 Request

Field name	Description	Required	Data type
trxId	Unique ID for transaction	Yes	String(30)
merId	Unique ID for Merchant	Yes	String(10)
merTrxId	Transaction code generated by merchant	Yes	String(50)
type	Type of installment cancel: +) 1: Request cancel installment (IS cancel) +) 2: Request Cancel installment (IS cancel) and Refund (refund) Note: Only applicable for Installment (IS) payment method	Yes	String(2)
cancelMsg	The reason of canceling transaction	No	String(100)
timeStamp	timeStamp	Yes	String(13)
cancelServiceIp	IP of canceling server	No	String(15)
cancelUserId	ID of canceling user	No	String(30)
cancelUserIp	IP of canceling user	No	String(15)
cancelUserInfo	Information of canceling user	No	String(100)
merchantToken	Sha256(timeStamp + merTrxId + trxId + merId + type + encodeKey)	Yes	String(255)
cancelPw	Merchant's canceling password (encoded) Password shared with Refund	Yes	String(255)



10.2 Response

Field name	Description	Required	Data type
trxId	Code of canceled transaction	No	String(30)
invoiceNo	Order code	No	String(40)
cancelTrxId	Code of canceling transaction (Generated by megapay)	No	String(30)
resultCd	Result of canceling transaction: 00_000: Request Cancel Installments Success 99: Request for Cancellation installments "Waiting for approval" Others: fail	Yes	String(6)
resultMsg	Message result of canceling transaction	Yes	String(100)
trxDt	Transaction date yyyyymmdd	No	String(8)
trxTm	Transaction time hh24miss	No	String(6)
timeStamp	Time stamp	No	String(13)
description	Description	No	String(100)
amount	The end user installment fee is received back	No	String(12)
merTrxId	Transaction code generated by Merchant	No	String(50)
merchantToken	Sha256(resultCd +timeStamp + merTrxId + trxId + merId + amount + encodeKey)  Note : In case of failed merchant token verification, please call inquiry function to get the exact status of the transaction	No	String(255)
payMessage	CC_ISAN, ignore this value	No	String(7)

Message template

- Successful IS cancel:

```
{
  "trxId": "SITNAPAS01IC201908231318287662",
  "invoiceNo": "OrdNo20190823082433",
  "cancelTrxId": "SITNAPAS01IC201911111526356652",
  "resultCd": "00_000",
  "resultMsg": "SUCCESS",
  "trxDt": "20190823",
  "trxTm": "131846",
  "description": "fdsafsd",
  "amount": "1000",
  "merchantToken": "058404a42a6e412970bdd5efa9935362dfcdfa395e97e131c067e8a5c3b0b662",
  "merTrxId": "SITNAPAS0120190823082433",
  "payMessage": "CC_ISAN",
  "timeStamp": "20190823082433"
}
```

- IS cancel awaiting approval:

```
{
  "trxId": "SITNAPAS01DC201908161421387007",
  "invoiceNo": "OrdNo20190816092739",
  "cancelTrxId": null,
  "resultCd": "99",
  "resultMsg": "The transaction has been confirmed by installments, so it is necessary to wait for approval",
  "trxDt": "20190816",
  "trxTm": "142138",
  "description": "fdsafsd",
  "amount": "500000",
  "merchantToken": "b4a8d1d78a80e139ebbf65196cf18c96a32609c6d39838f1733a783858f82667",
  "merTrxId": "MERTRXID20190816092739",
  "payMessage": "CC_ISAN",
  "timeStamp": "20190816092739"
}
```

- Failed IS cancel:



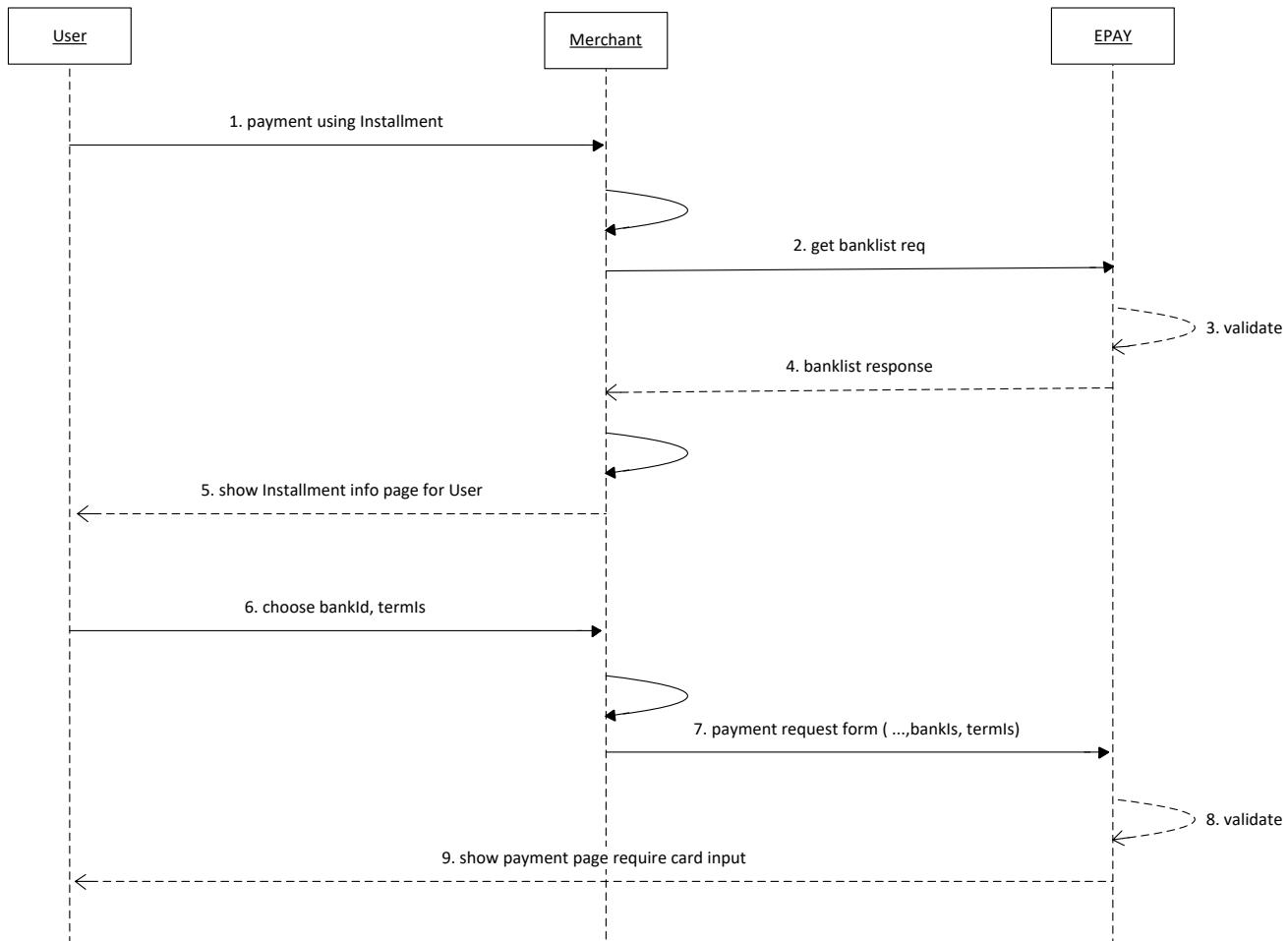
```
{"trxId":null,"invoiceNo":null,"cancelTrxId":null,"resultCd":"CC_110","resultMsg":"Invalid transaction number.", "trxDt":null,"trxTm":null,"description":null,"amount":null,"merchantToken":"9368f3783210175e5a457ac4c2cf0ba759cee9d3c61da8a5c38cbed33715ee48","merTrxId":null,"payMessage":"CC_ISAN","timeStamp":"1566294563371"}
```

11 API to get installment information



This function only applies to **Installment payment method** when merchant wants to display installment information on its page

11.1 Integration flow



Description of the steps:

Step 1: User selects product/service at merchant's page and makes payment by installment method.



Step 2: Merchant requests API to get installment information (in section 11.2) including: MID + Product value + Signature...

Step 3: EPAY Receive request and validate information.

Step 4: EPAY returns the list of banks that support installment payments with some information such as term and fee information, etc.

Step 5: Merchant receives installment information and displays it on the screen asking the user to choose a bank and installment term.

Step 6: User selects bank, installment period, check installment information. Press the Next button.

Step 7: Merchant requests payment (in section 5.1), sends additional information to EPAY including: installment bank (bankCode) and installment term (termIs)

Step 8: EPAY receives request and validate information.

Step 9: Display the payment window for user to enter the card information.

The user's payment process is same as international card transaction.

11.2 Request

URL: https://sandbox.megapay.vn/pg_was/api/listbankIS.do

Method: POST

Data type: **application/x-www-form-urlencoded**

Field name	Description	Required	Data type
merId	Unique ID for Merchant Example: [String] "MEGAPAYTEST"	Yes	String(10) - Fixed
amount	Transaction amount	Yes	String (12)
merchantToken	Merchant's signature: used to authenticate merchant's transaction. - Formula: Sha256(timeStamp + merTrxId + merId + amount + encodeKey)	Yes	String(255)
timeStamp	timeStamp Format: yyyyMMddHHmmss	Yes	String(13)
merTrxId	Unique transaction ID generated by merchant (This trans ID will be used for inquiry transaction function) Format: merId + unique number	Yes	String(50)

11.3 Response

Data type: Json/application



Field name	Description	Required	Data type
merId	Merchant ID on MegaPay system	Yes	String(10)
merTrxId	Transaction ID generated by Merchant	Yes	String(50)
resultCd	Transaction's result (refer to error code table)	Yes	String(6)
resultMsg	Message result of canceling transaction	Yes	String(512)
userFee	Payment fee of end user	Yes	String
merFee	Payment fee of merchant	Yes	String
amount	Transaction amount	Yes	String(12)
merchantToken	<p>Signature: it was used to authenticate merchant's transaction.</p> <p>- Formula: Sha256(resultCd + merId + amount + userFee + merFee + data + encodeKey)</p>	Yes	<p>String(255)</p>  <p>In case of failed merchant token verification (mismatched or exception), let's call inquiry function again to get the exact status of transaction</p>
data	<p>List of banks supporting installment payment data *encrypted 3DES Decryption Key: first 24 characters of encodeKey</p>	Yes	String

The **data** field after decoding is in the form of a multi-bank json array, including each element as a bank object containing the following information:

Field name	Description	Data type
bankId	Bank code	String
listDuration	List of installment terms	Array json

The **listDuration** field is a json array, each element is an installment term object containing the following information:

Field name	Description	Data type
termIs	Available installment terms (unit is month)	String
userFeeIs	Installment conversion fee that enduser bears	String
merFeeIs	Installment conversion fee that merchant bears	String
amountIs	Total payment amount (including user payment fee and user installment fee) according to the term of each bank and merchant configuration	String
firstAmount	The amount of the 1 st payment	String
nextAmount	The amount of the 2 nd payment	String



12 Delete unused token



This function only applies to when the end user account is closed or deleted, it is necessary to delete this end user token to ensure that the unused token is deleted.

URL: https://sandbox.megapay.vn/pg_was/deleteTokenAPI.do

Method: POST

12.1 Request

Field name	Description	Required	Data type
merId	Unique ID for Merchant	Yes	String(10)
payType	Payment method: – IC : Credit card (Visa/master/JCB...) – DC : Domestic card (ATM)	Yes	String(2)
payToken	Tokenization is not used, needs to be deleted Encrypted by 3DES Encrypted Key: 24 last character of encodeKey Note: This payToken is the same as the payToken used for payment in the case of payOption with PAY_WITH_TOKEN	Yes	String(100)
merchantToken	Signature: used to authenticate the merchant's request. Formula: Sha256(merId + payType + payToken + encodeKey)	Yes	String(255)

12.2 Response

Field name	Description	Required	Data type
resultCd	Result of delete token: 00_000: Request delete token success Others: fail	Yes	String(6)
resultMsg	Message result of delete token	No	String(100)
merId	Unique ID for Merchant	Yes	String(10)
payType	Payment method: – IC : Credit card (Visa/master/JCB...) – DC : Domestic card (ATM)	No	String(8)



merchantToken	Signature: authentication result returned by Megapay. Formula: Sha256(resultCd + merId + payType + encodeKey)	No	String(255)
---------------	--	----	-------------

Message template

- Success delete token:

```
{ "resultCd": "00_000", "resultMsg": "SUCCESS", "merId": "EPAY000001", "payType": "IC", "merchantToken": "058404a42a6e412970bdd5efa9935362dfcdfa395e97e131c067e8a5c3b0b662" }
```

- Failed delete token :

```
{ "resultCd": "IC_126", "resultMsg": " Token does not exist", "merId": "EPAY000001", "payType": "IC", "merchantToken": "9368f3783210175e5a457ac4c2cf0ba759cee9d3c61da8a5c38cb33715ee48" }
```

13 API to check connection information



This API is for the purpose of verifying the merchant's connection information

URL: https://sandbox.megapay.vn/pg_was/checkInfoIntegrateMerId.do

Method: POST

13.1 Request

Field name	Description	Required	Data type
merId	Unique ID for Merchant	Yes	String(10)
encodeKey	<ul style="list-style-type: none">- Key for integrating payment flow, issued by MGP to merchant- This key can be change on MMS	Yes	String(255)
hashKeyRefund	<ul style="list-style-type: none">- Key for integrating refund flow, issued by MGP to merchant- This key can be change on MMS	Yes	String(255)
timeStamp	timeStamp <ul style="list-style-type: none">- Format: yyyyMMddHHmmss	Yes	String(14)
hash	Merchant's signature: it was used to authenticate merchant's transaction. Formula: Sha256(merId + encodeKey + hashKeyRefund + timeStamp)	Yes	String(255)



13.2 Response

Data type: Json/application

Field name	Description	Required	Data type
resultCd	00_000: MGP connection is successful Others: fail	Yes	String(6)
resultMsg	Detailed result message	No	String(100)
merId	Unique ID for Merchant	Yes	String(10)
status	Information check result status: 0: This merId does not exist on MGP's system 1: The information is valid, the information exists in the MGP 2: Wrong endcodeKey or hashKeyRefund 3: Merchant is paused or locked	Yes	String(10)
hash	Signature: authentication result returned by Megapay. Formula: Sha256(resultCd + merId + status)	No	String(255)

14 Capture Request API



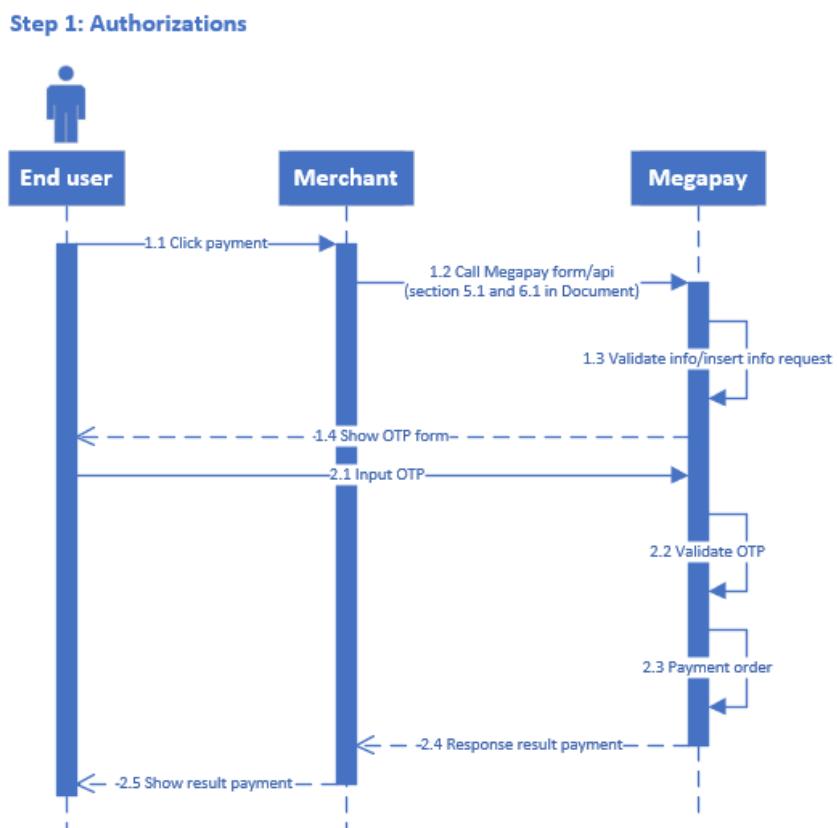
This function only applies to **International card payment method (IC, IS)**

14.1 Integration flow

Dual-message transaction (two steps)

Step1: Authorizations (*Request to temporarily hold the amount for payment*)

- Request as normal payment at section 5.1 and 6.1, but need to pass one more parameter “**authorizations**” with value “**1**”
- Note: When a transaction is “authorized” only, the transaction is **NOT** completed, merchant should **NOT** provide products or services before receiving result from capture request below. Authorized only transactions are not completed transactions; hence will **NOT** be settled to merchants.
- Process flow:





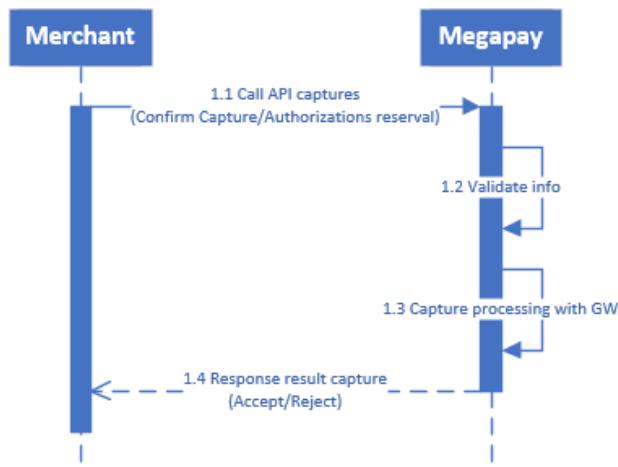
Description of the steps:

- 1.1 : End user chooses payment of merchant's website
- 1.2 : Merchant makes a call to MGP: payment with request form at section 5.1 or pay with Token API direct of international card at 6.1
- 1.3 : MGP validates the merchant's information, if the information is correct, then save the merchant's request information
- 1.4 : MGP show OTP form of card issuer returned to End user
- 2.1 : End User performs OTP input
- 2.2 : Card issuer validates OTP
- 2.3 : MGP process transaction, send request to authorize the transaction (Request to hold the amount need payment)
- 2.4 : MGP returns payment results to Merchant
- 2.5 : Merchant shows payment results to End user

Step2: Captures (Complete the authorized transaction)

- This step must be taken if payment request to MGP (form – 5.1 /API – 6.1) was sent with a parameter “**authorization**” as “**1**”
- **Note:** When a transaction is “authorized” only, the transaction is **NOT** completed, and if a capture request is not sent within 7 days after the authorization, temporary hold on user’s account could be released depending on card issuers and capture request could fail. To ensure that transaction is completed, sending capture request within 7 days is recommended.
- Process flow:

Step 2: Captures





Description of the steps:

- 1.1 : Merchant requests capture or authorizations reversal
- 1.2 : MGP validates info request of Merchant
- 1.3 : MGP processes capture with card issuers
- 1.4 : MGP returns results to Merchant

14.2 Request

URL: https://sandbox.megapay.vn/pg_was/api/paymentCaptures.do

Method: POST

Data type: Json/application

Field name	Description	Required	Data type
trxId	Unique transaction ID of authorized transaction that merchants want to complete (trxId that MGP returned for the authorization request (5.1 & 6.1))	Yes	String(30)
merId	Unique ID for Merchant	Yes	String(10)
amount	Amount of previously authorized transaction Sample: [String] "500000"	Yes	String (12)
type	Transaction type: +) 1: Capture (complete the previously authorized transaction) +) 2: Authorizations reversal (cancel previously authorized transaction - release temporary hold on user's account)	Yes	String(1)
timeStamp	timeStamp Format: yyyyMMddHHmmss	Yes	String(13)
note	Information to note Example, the reason for wanting to make this request	No	String (300)
merchantToken	Formula: Sha256(trxId + merId +amount + type + timeStamp + encodeKey)	Yes	String(255)

14.3 Response

Data type: Json/application

Field name	Description	Required	Data type
trxId	Unique ID for transaction, original transaction code of MGP	Yes	String(30)
merId	Merchant ID on MegaPay system	Yes	String(10)
amount	Amount requested for a capture Sample: [String] "500000"	Yes	String (12)
type	Transaction type: +) 1: Capture (complete the previously authorized transaction) +) 2: Authorizations reversal (cancel previously authorized transaction - release temporary hold on user's account)	Yes	String(1)

resultCd	Result of transaction 00_000 : Successful Others: fail	Yes	String(6)
resultMsg	Result description	Yes	String(100)
merchantToken	<p>Sha256(resultCd + trxId + merId + amount + type + encodeKey)</p>  <p>In case of failed merchant token verification (mismatched or exception), call inquiry function to get the exact status of transaction</p>	No	String(255)

15 Appendix

15.1 Payment method list

Payment method name	Payment method code
Pay by ATM card	DC
Pay by credit card (Visa/master/JCB...)	IC
Pay by installment (credit card issued domestic)	IS
Pay by eWallet	EW
Pay by Deposit code	VA
Pay by VNPayQR	QR

15.2 Error code list

STT	Result code	Cause (English)
1	00_000	Successful Note: With pay_type = VA (Dcode), this transaction enduser has successfully transferred the amount
2	00_005	The transaction has been mapped to the Dcode account successfully, but the enduser has not paid yet Note: Only applies to pay_type = VA (Dcode)
3	FL_900	Connection error.
4	FL_901	Socket connection error.
5	FL_902	Some error happened while processing.
6	FL_903	Socket timeout exception.
7	OR_101	MerID is invalid or Merchant is not registered. Contact to Customer Service Center for further information.
8	OR_102	This Payment type doesn't exist or currently not activated. Contact Customer Service for further information.



9	OR_103	Currency Code was undefined. Please check your request parameter and make sure [currencyCode] is defined.
10	OR_104	buyerCity was undefined. Please check your request parameter and make sure [buyerCity] is defined.
11	OR_105	InvoiceNo was undefined. Please check your request parameter and make sure [invoiceNo] is defined.
12	OR_106	Goods Name was undefined or wrong format. Please check your request parameter and make sure [goodsNm] is defined.
13	OR_107	Buyer first name or last name was undefined. Please check your request parameter and make sure [buyerFirstNm] and [buyerLastNm] is defined.
14	OR_108	Buyer Phone Number was undefined. Please check your request parameter and make sure [buyerPhone] is defined.
15	OR_109	Buyer Email Address was undefined. Please check your request parameter and make sure [buyerEmail] is defined correctly.
16	OR_110	Callback URL was undefined. Please check your request parameter and make sure [callbackUrl] is defined.
17	OR_111	Notification URL was undefined. Please check your request parameter and make sure [notiUrl] is defined.
18	OR_112	Payment Amount is invalid. Amount should only number and do not include decimal.
19	OR_113	Invalid Merchant Token. Contact Customer Service for further information.
20	OR_114	Payment Amount has to be greater than 0. Please check your request parameter and make sure [amount] is defined.
21	OR_115	The flag field which was used to identify if a merchant will be checked duplicate order no is null.
22	OR_116	Invoice no is duplicated
23	OR_117	Duplicate Merchant transaction ID (merTrxId)
24	OR_118	Error because of one of the following causes: +) Request domain was undefined +) MerchantId sent by merchant is null +) MerchantId is mismatch (Inquiry transaction function)
25	OR_120	Merchant's status error (Merchant is not active)
26	OR_122	Merchant Transaction ID is empty or wrong format
27	OR_123	Error of merchant is not defined in the system.
28	OR_124	Merchant's status error (Merchant is not active)
29	OR_125	Merchant is not registered this payment method or Cybersource settle time is not defined.
30	OR_126	PG Type is not set.
31	OR_127	Check Merchant's <u>applied</u> limit error.
32	OR_128	Over limit amount error.
33	OR_130	The field used to identify merchant is online or offline is not defined. Please check your request parameter and make sure [merType] is defined



34	OR_131	This Online Merchant type is currently not activated.
35	OR_132	This Offline Merchant type is currently not activated.
36	OR_133	Contract's information is not defined.
37	OR_134	Invalid amount
38	OR_140	Transaction does not exist
39	OR_141	Buyer address is required
40	OR_142	Buyer state is required when buyer country is 'us' or 'ca'
41	OR_143	Buyer country is required
42	OR_147	description is invalid
43	OR_148	timeStamp is not allowed to be empty or invalid
44	OR_150	The system doesn't support tokenization for this payment method
45	OR_151	userId is not allowed to be empty or is invalid
46	OR_152	userId is invalid or empty
47	OR_153	payOption is invalid
48	OR_154	Not support merchant ON 3DS
49	OR_155	Only support payment type is International card
50	OR_156	Only support merchant using token with International card
51	OR_157	Not support International Card payment
52	OR_158	Not support ATM Card payment
53	OR_159	Not support eWallet payment
54	OR_160	Transaction timeout
55	OR_161	Bank maintenance
56	OR_162	Not support installment payment
57	OR_163	[type] is Invalid or NULL
58	OR_164	Order has already been processed successfully, please make another order
59	OR_165	Order is under processing. Please try again after 30 minutes in maximum



60	OR_166	No url configured
61	OR_167	Card information has been deleted because there has been no transaction for a long time. Please re-enter the previously saved card information.
62	OR_168	Card you have saved previously has expired. Please return to the purchase page, delete the saved card and pay with a new card. (User Information >> Payment Management >> Select Card to Delete >> Delete Card)
63	OR_169	The card number you just entered does not match the card number registered with the merchant. Please enter the card number registered:
64	OR_170	Payment link does not exist or has expired
65	OR_171	Payment link is being processed
66	OR_172	WindowType is invalid
67	OR_173	User Language is invalid
68	OR_174	Invalid Merchant installment declaration
69	OR_175	Invalid [_Field_] field length. Please check again
70	DC_101	Error occurred while checking information sending to or receiving from NAPAS
71	DC_102	Transaction Id is not defined
72	DC_103	Transaction already exist. Please make new transaction.
73	DC_104	Invoice no is null. Please make sure that [invoiceNo] was defined already
74	DC_105	Data is null error.
75	DC_110	Payment type is undefined. Contact <u>Megapay</u> for further information.
76	DC_112	Error occurred while inserting or updating data in the tables related to ATM transaction.
77	DC_113	Error occurred while updating email transaction.
78	DC_114	Error when saving data into notify transaction table.
79	DC_117	Transaction's information haven't registered yet. Please check again.
80	DC_119	Server is busy. Please kindly try again in few minutes.
81	DC_120	Payment Success but token is not created.
82	DC_122	Partner transaction Id is null
83	DC_123	Partner transaction Id is invalid
84	DC_124	Invalid amount
85	DC_125	Invalid currency



86	DC_126	Wrong CVV
87	DC_127	Transaction value is out of limit range set by your bank
88	DC_128	Invalid expiration date
89	DC_129	Invalid issue date
90	DC_130	value does not meet minimum limit set by bank
91	DC_131	Expired napas Session
92	DC_132	Transaction not supported
93	DC_133	The card inputted is not promotion program
94	DC_134	Order is expired
95	DC_135	Authentication failed 3ds
96	IC_101	Transaction failed. Please check card information and try again.
97	IC_102	Transaction Id is not defined
98	IC_103	Transaction already exist. Please make new transaction.
99	IC_104	Invoice no is undefined (null). Please check field named [invoiceNo] again
100	IC_105	Merchant's card information is not defined
101	IC_107	Error occurred while connecting to CyberSource
102	IC_110	Paytype or merchant id is missing
103	IC_112	Error occurred while inserting data into tables related to credit card transaction
104	IC_113	Error occurred while updating email transaction table
105	IC_115	Invalid MID, Merchant is not registered. Contact <u>Megapay</u> for further information.
106	IC_117	Transaction's information is not registered
107	IC_121	Merchant is inactive
108	IC_122	payToken is in wrong format or empty
109	IC_123	Merchant doesn't support payment with Tokenization with international card
110	IC_124	Token was not found
111	IC_125	Token locked



112	IC_126	Token deleted
113	IC_127	Card type (cardTypeValue) is invalid
114	IC_128	Signature (cardTypeToken) is invalid
115	IC_129	Card type (cardTypeValue) is not support
116	IC_130	Merchant has not created fee
117	IC_131	Not allowed to accept international card (domestic)
118	IC_132	Not allow to accept international card (overseas)
119	IC_133	Decrypting token error
120	IC_134	Fraud card
121	IC_136	Discount Code is invalid or not applied for this Card
122	IC_137	The entered card is not supported. Please make the transaction again by another card
123	IC_138	Decline - Stolen or lost card.
124	IC_139	Decline - Invalid Card Verification Number (CVN).
125	IC_140	Decline - Processor failure.
126	IC_141	Decline - The Pinless Debit card use frequency or maximum amount per use has been exceeded
127	IC_142	Invalid postal code
128	IC_143	Invalid buyer country
129	IC_144	Invalid ship country
130	IC_145	Missing field buyer country
131	IC_146	Not receiving response events from Cardinal.
132	IC_147	Transation Capture/Authorization Reversal is processing or success.
133	IC_148	MerchantToken is not match.
134	IC_149	Inactive not authorized for online transactions
135	VA_101	Connect to VA system is fail or transaction id is not defined
136	VA_102	VA transaction has already existed
137	VA_103	Merchant id information is missing



138	VA_104	Error occurred while inserting data into VA trans table
139	VA_105	Error occurred while inserting data into trans result table.
140	VA_106	Error occurred while searching transaction or transaction doesn't exist
141	VA_107	Error occurred while inserting data into merchant notification table
142	VA_109	Condition of receiving money is wrong (should equal to 03)
143	VA_110	Effective date is wrong
144	VA_111	Expire date is wrong
145	VA_112	Transaction is invalid
146	VA_113	Cannot find Bank issuing Deposit code
147	CC_101	Transaction id is not generated.
148	CC_102	Invalid MID or Merchant is not registered. Contact Customer Service for further information.
149	CC_109	Merchant in the not active status
150	CC_110	The transaction has not been registered
151	CC_111	Error occurred when canceling amount less than or equal 0 or canceling amount not equal transaction amount (In case of comprehensive cancelation)
152	CC_112	The transaction needs to be canceled is not found.
153	CC_113	The transaction has been canceled fully
154	CC_114	This payment method is currently not activated with merchant or insert notification data has been failed
155	CC_115	Merchant's token is invalid
156	CC_116	Canceling amount must equal payment amount.
157	CC_117	Amount of cancelation/refunding is invalid (Format number exception)
158	CC_118	It has been canceled.
159	CC_119	The amount of money you entered is larger than the payment amount or the remaining payment amount is less than 0.
160	CC_121	Error occurred when updating transaction's information
161	CC_122	Error occurred when inserting partial cancelation information
162	CC_124	Error occurred when inserting data after canceling
163	CC_125	Transaction result registration error.



164	CC_126	Error occurred when querying data
165	CC_127	Partial cancelation flag or status is invalid.
166	CC_128	Cancel message is not defined
167	CC_130	The amount of money you want to cancel is <u>smaller</u> than the transaction's payment amount.
168	CC_131	Error occurred when inserting data into cancelation transaction table
169	CC_132	Update transaction history error.
170	CC_133	Bank connection error.
171	CC_135	Cancel password is mismatched.
172	CC_136	Canceling function is unavailable with this merchant. Please Contact to <u>Meagepay</u> .
173	CC_141	Currently cannot make a partial refund for this transaction, please wait until tomorrow
174	CC_143	This transaction must refund all
175	PG_ER1	Transaction Failed.
176	PG_ER2	Card's information is wrong
177	PG_ER3	Transaction is failed - Timeout
178	PG_ER4	Transaction is failed
179	PG_ER5	Customer canceled transaction
180	PG_ER6	System error, please contact to Megapay's Admin for supporting
181	PG_ER7	Card number is invalid
182	PG_ER8	Publish/Expire date is invalid
183	PG_ER10	Buyer address is wrong
184	PG_ER11	Card has not been configured Payer Authentication yet
185	PG_ER12	Buyer last name or first name is wrong
186	PG_ER13	Buyer city/state is wrong
187	PG_ER16	OTP is wrong
188	PG_ER17	Card information has not been approved yet, please contact to issuing bank to be supported
189	PG_ER18	Card expired or locked



190	PG_ER19	The amount of money is not enough to make a payment
191	PG_ER20	The amount of money of transaction is not within the allowed limit
192	PG_ER21	Card has not been activated or signed up for online payment yet
193	PG_ER22	Card holder name is wrong
194	PG_ER23	Issuing bank denied the transaction
195	PG_ER25	The transaction was denied by fraud management system
196	PG_ER26	Data is invalid or empty
197	PG_ER28	Issuing bank is postponing this transaction. Please try again later
198	PG_ER29	Transaction failed because customer is in blacklist
199	PG_ER30	Transaction failed – Cannot authenticate the customer
200	PG_ER31	Transaction exceeds daily limit set by bank
201	PG_ER32	Transaction value exceeds maximum limit set by bank
202	PG_ER42	OTP time out (if you are charged, it will be refunded)
203	PG_ER43	Issuing bank is busy. Please try again
204	ZL_1	Pending with zalopay
205	ZL_2	User error with zalopay
206	ZL_3	Merchant error with zalopay
207	ZL_4	System error with zalopay
208	ZL_5	Unknown error with zalopay
209	ZL_6	User error and system error with zalopay
210	ZL_7	Bank error with zalopay
211	ZL_8	Other error with zalopay
212	MM_1	Empty accessKey or partnerCode
213	MM_12	Duplicated requestID
214	MM_14	Partner is not activated
215	MM_2	OrderId is in wrong format



216	MM_29	System maintenance. Please try in few minutes
217	MM_32	Transaction was purchased
218	MM_33	Transaction cannot be refunded
219	MM_34	Transaction refunded
220	MM_36	Expired transaction
221	MM_37	Capset exceeded
222	MM_38	Insufficient funds
223	MM_4	Amount is invalid, should be between 1,000VND and 20,000,000 VND
224	MM_44	Service does not support your request
225	MM_49	Order cancelled by user
226	MM_5	Signature is wrong. Check raw signature before signed
227	MM_58	Transaction does not exist
228	MM_59	Error parsing body to Json object
229	MM_6	OrderId exists
230	MM_63	Pay by bank source failed
231	MM_7	Pending transaction
232	MM_80	User failed authentication
233	MM_9043	User does not link bank account
234	MM_99	Error undefined
235	MC_1	Moca failed
236	MC_5	Moca invalid_request
237	MC_6	Moca unauthorized_client
238	MC_7	Moca access_denied
239	MC_8	Moca unsupported_response_type
240	MC_9	Moca invalid_scope
241	MC_10	Moca temporarily_unavailable



242	MC_11	Moca interaction_required
243	MC_12	Moca login_required
244	MC_13	Moca account_selection_required
245	MC_14	Moca consent_required
246	MC_15	Moca invalid_request_uri
247	MC_16	Moca invalid_request_object
248	MC_17	Moca request_not_supported
249	MC_18	Moca request_uri_not_supported
250	MC_19	Moca registration_not_supported
251	IS_001	Bank or term of installment not supported
252	VT_01	Transaction fail from Viettel Money or Bank, if user has been deducted money, will be refunded within 7-15 days. Please contact CS center 18009000 for support
253	VT_02	Wrong password/PIN/OTP or expired OTP
254	VT_03	Viettel Money account has been inactivated/locked/cancelled. Please contact CS center 18009000 for support
255	VT_04	Cannot find the linked card information or card link has expired. Please unlink and re-link
256	VT_05	Phone number doesn't match / didn't register with card number
257	VT_06	Amount or count of transaction exceeded monthly limit
258	VT_07	Didn't input phone number or payment amount. Please contact CS center 18009000 for support
259	VT_08	Cannot find subscriber information/account cardholder information/ID number or the information doesn't match. Please check again or contact CS center 18009000 for support
260	VT_09	Cannot find phone number to send OTP or cannot send OTP. Please contact CS center 18009000 for support
261	VT_10	Viettel Money account hasn't been linked for automatic payment, please check again before implementing transaction again
262	VT_11	Viettel Money account has been linked for automatic payment, please check again before implementing transaction again
263	VT_12	You are not eligible to use Mobile Money service. Please contact CS center 18009000 for support
264	VT_13	Transaction failed. Account needs to change PIN before making Online payment, please access Viettel Money App to check.

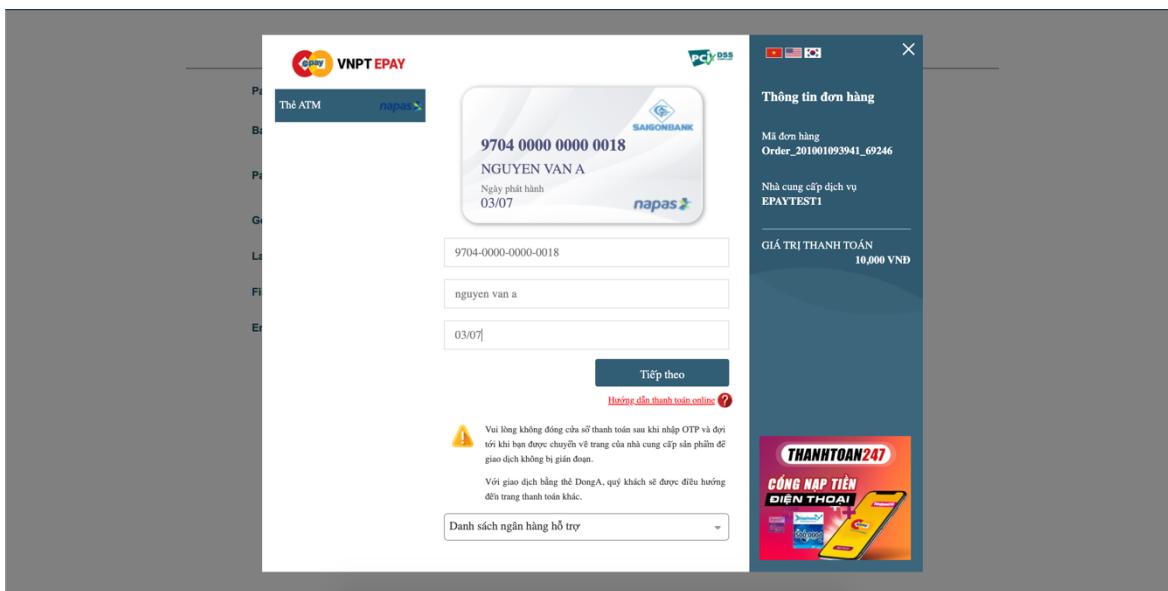
15.3 List bank

#	BankCode	Bank Name
1	VCCM	Ngân hàng Bản Việt
2	NASM	Ngân hàng Bắc A
3	KLBM	Ngân hàng Kiên Long
4	PVCM	PVcombank
5	TPBM	TienphongBank
6	PGBM	PG bank
7	TCBM	TechcomBank
8	STBM	Sacombank
9	OJBM	OceanBank
10	NCBM	Ngân hàng Quốc Dân
11	HDBM	HD Bank
12	VABM	Việt Á Bank
13	VTBM	Vietinbank
14	VPBM	VP Bank
15	ABBM	AB Bank
16	SEAM	SeaBank
17	SCBM	Ngân hàng TMCP Sài Gòn (SCB)
18	ACBM	ACB
19	VARM	Agribank
20	BVBM	Ngân hàng Bảo Việt
21	BIDM	BIDV
22	DABM	Dong A Bank
23	EIBM	EximBank
24	GPBM	GP Bank
25	LPBM	Ngân hàng Bưu Điện
26	MBKM	MB
27	MSBM	MSB
28	NABM	Nam A Bank
29	IVBM	Ngân hàng TNHH Indovina
30	OCBM	Ngân hàng Phương Đông (OCB)
31	SHBM	SHB
32	VIBM	VIBank
33	PBVN	Ngân hàng TNHH MTV Public Việt Nam
34	VCBM	VietcomBank
35	VRBM	Ngân hàng Việt Nga
36	WRBM	Ngân hàng Woori Bank
37	SVBM	Ngân hàng Shinhan Việt Nam
38	SGBM	Ngân hàng TMCP Sài gòn Công thương
39	UOBM	Ngân hàng TNHH MTV United Overseas Bank

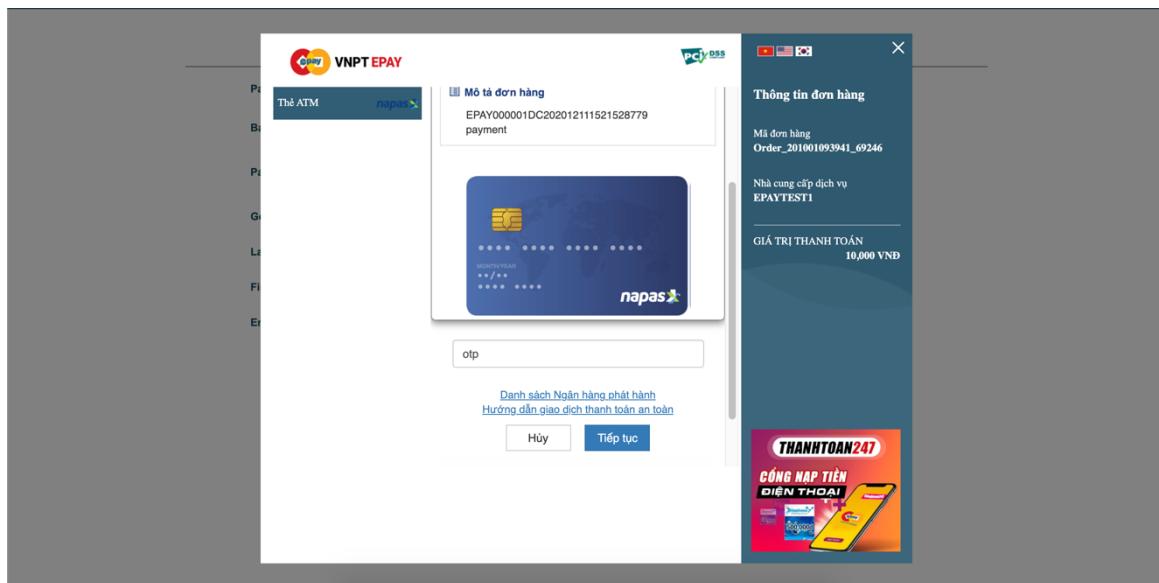
40	VBKM	Vietbank
41	CFCM	VietCredit Finance Joint Stock Company
41	FECM	FE credit (use with installment)
42	HMCM	Home credit (use with installment)
43	KEHN	Ngân hàng Keb Hana - CN HN
44	KEHC	Ngân hàng Keb Hana - CN HCM

16 Payment screen

16.1 Payment window of ATM card:

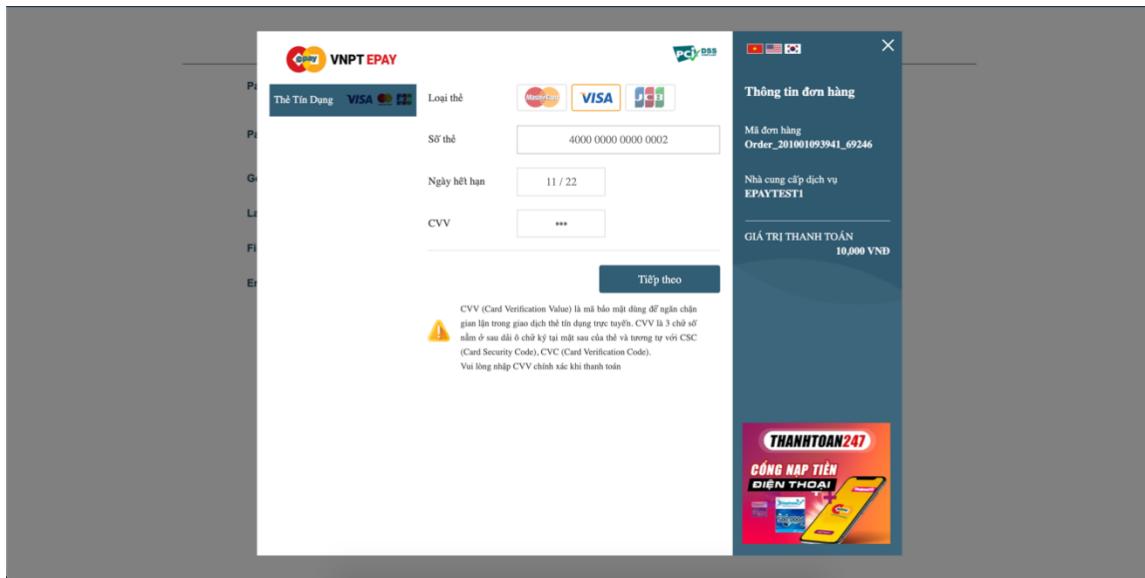


Card information input screen.

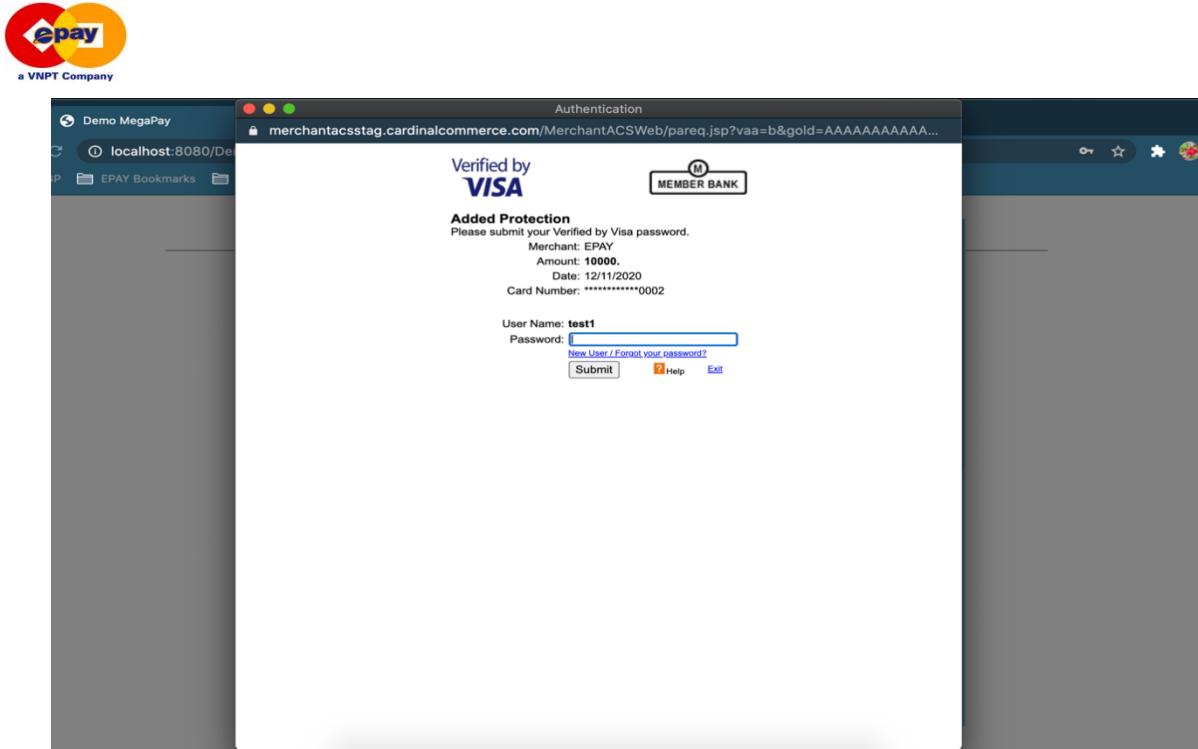


OTP authentication screen.

16.2 Payment window of International card:



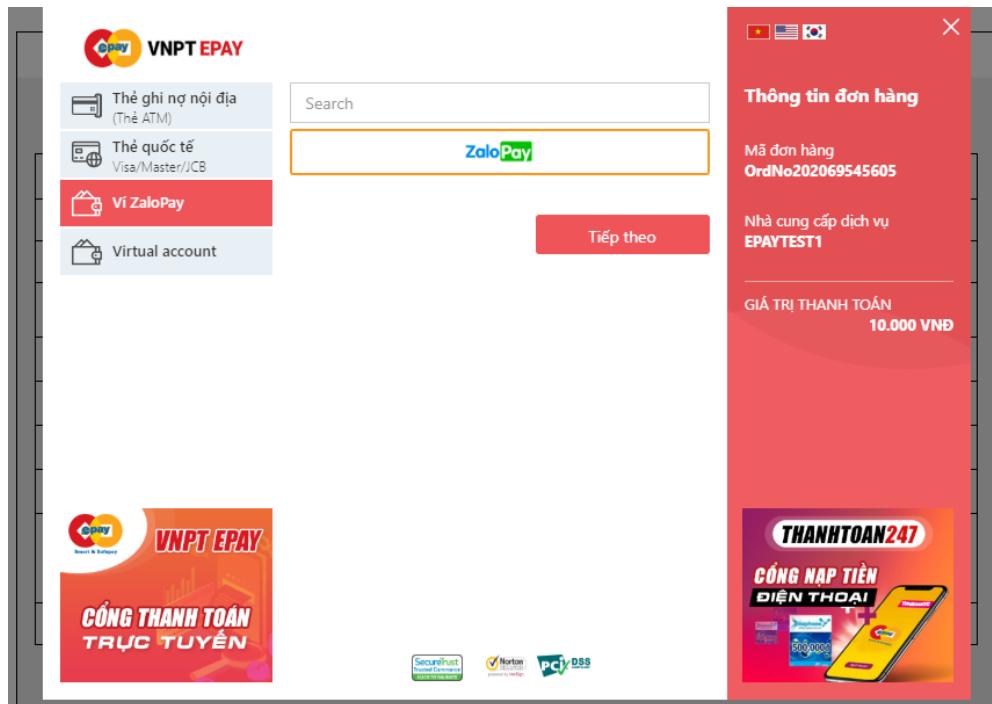
Card information input screen.

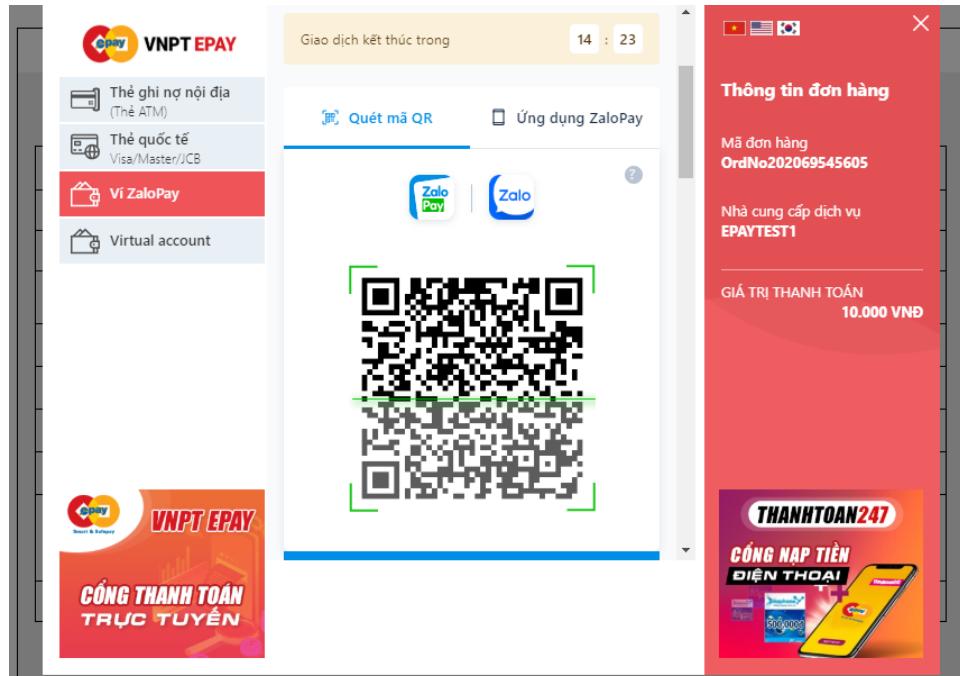


3Ds authentication screen.

17 Payment guide with ZaloPay

17.1 Payment window





17.2 Download ZaloPay app in Sandbox version

Download ZaloPay app (Sandbox version) using the link below:

Android: <https://dev.zalopay.com.vn/qrcode/public/app/android/merchant.html>

IOS: https://stcstg.zalopay.com.vn/ps_res/ios/enterprise/sandboxmer/5.3.0/install.html

After the download is completed:

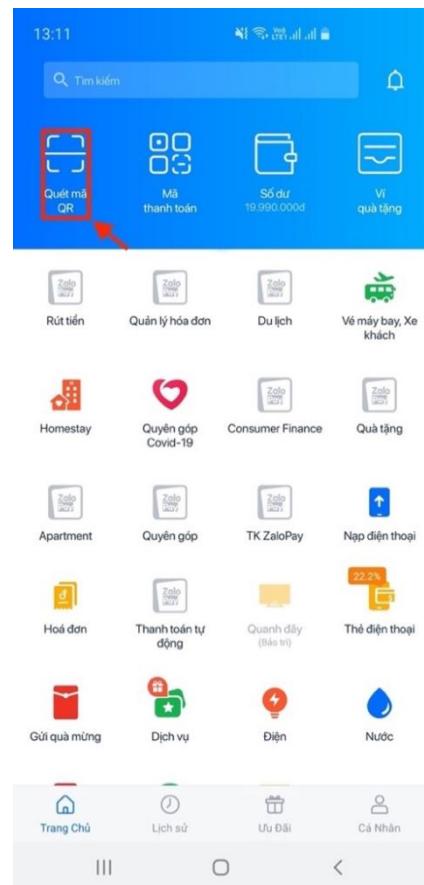
- Install the app in mobile
- Run the app and register an account by the mobile number already in use.
- Log in to Zalopay e-wallet

Make deposit into the registered Zalopay e-wallet by following the link below:

<https://docs.zalopay.vn/start/#A-II>

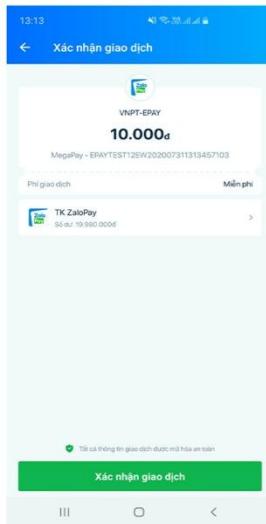
17.3 Use ZaloPay app (Sandbox version) & scan QR to pay

Step 1: Login to Zalopay e-wallet and tap “Scan QR” icon

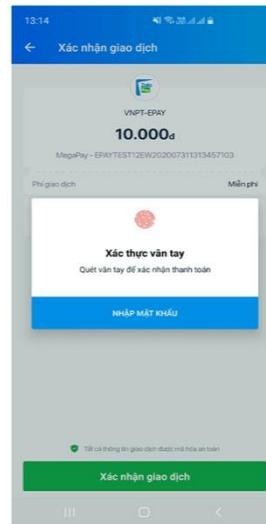


Step 2: Access mobile phone camera to scan the QR code in the payment screen (screen no.2 at section 14.1) then follow payment verification steps as requested by Zalopay e-wallet to complete payment.

Step 1: Transaction information after scanning the QR code



Step 2: Use password or fingerprint to confirm the transaction



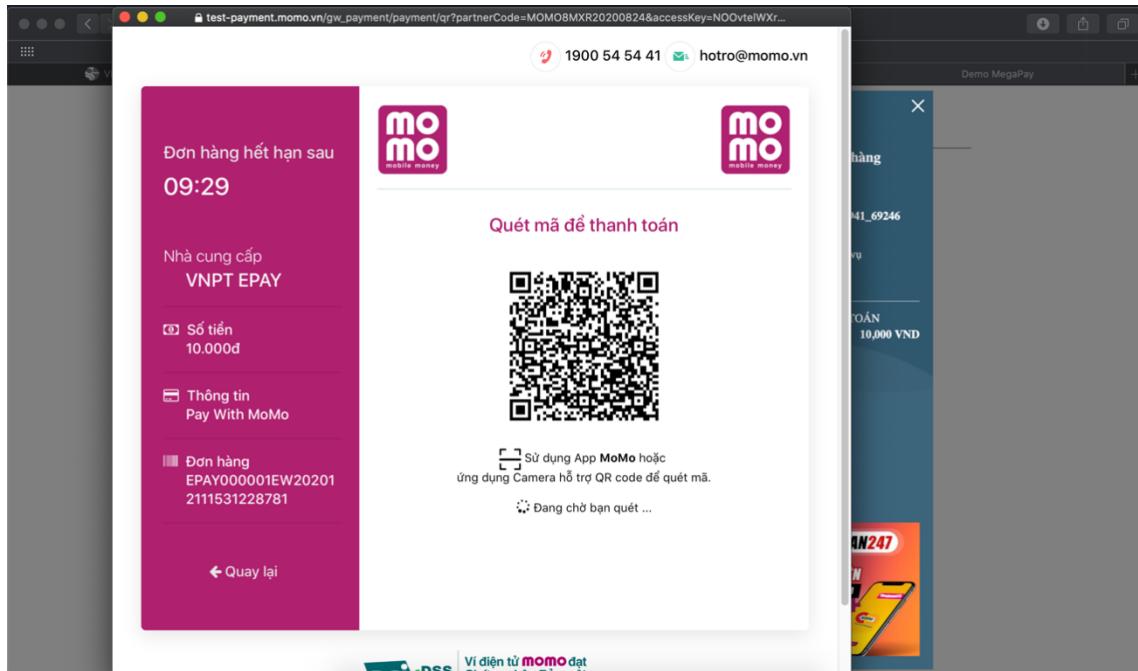
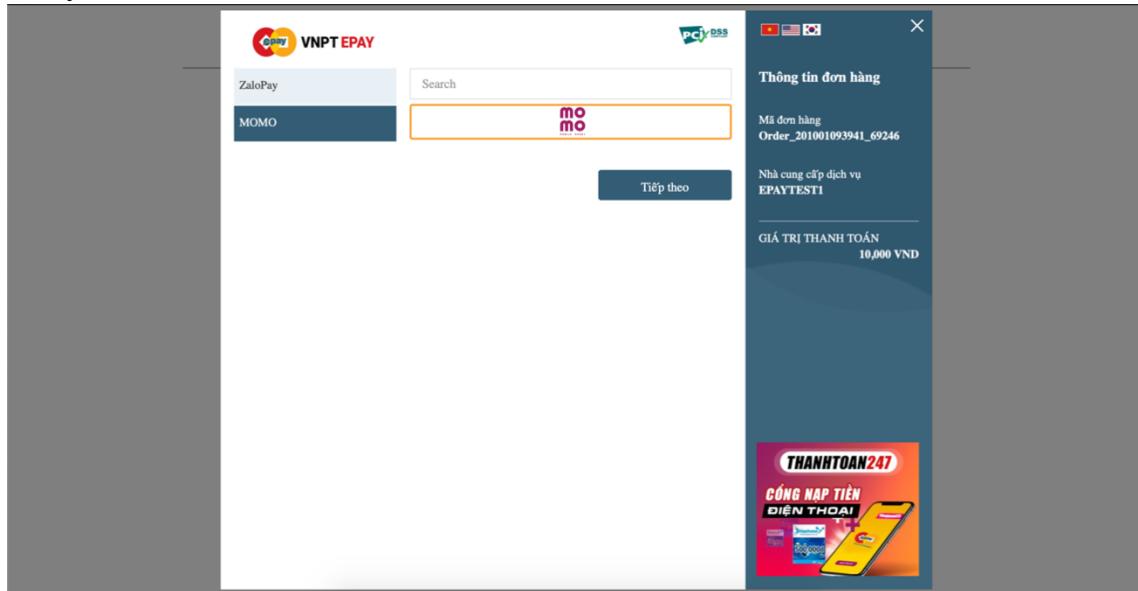
Step 3: The screen informs the successful transaction





18 Payment guide with Momo

18.1 Payment window



18.2 Download Momo app in Sandbox version

Download ZaloPay app (Sandbox version) using the link below:

<https://test-payment.momo.vn/download/>



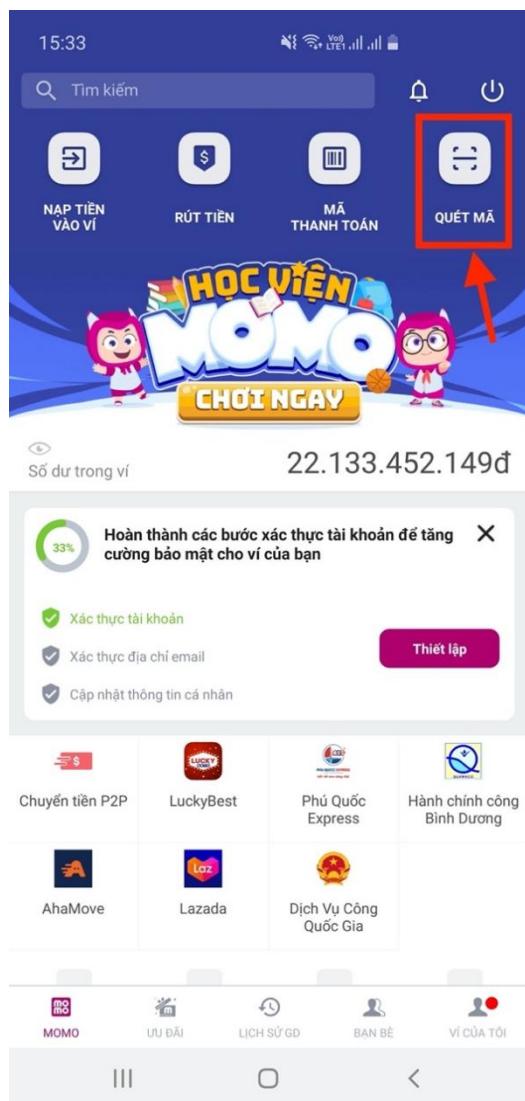
IOS: Need to setting up trust application: Settings > General > Profies & Device Management > Trust MoMo App.

After the download is completed:

- Install the app in mobile.
- Run and login the app using one of these mobile number below:
- 0917003003, 0917030000, 0917030003, 0917030030, 0917030300, 0917300300
- Password: 000000
- Authentication code (OTP): 000000

18.3 Use Momo app (Sandbox version) & scan QR to pay

Step 1: Login to Momo e-wallet and tap “Scan QR” icon:



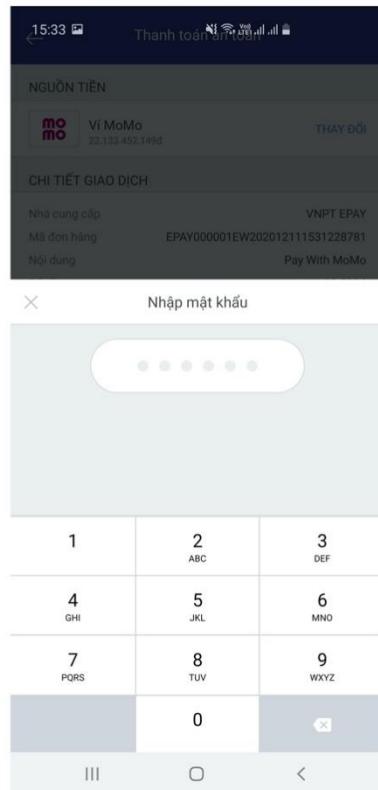


Step 2: Access mobile phone camera to scan the QR code in the payment screen (screen no.2 at section 15.1) then follow payment verification steps as requested by Momo e-wallet to complete payment.

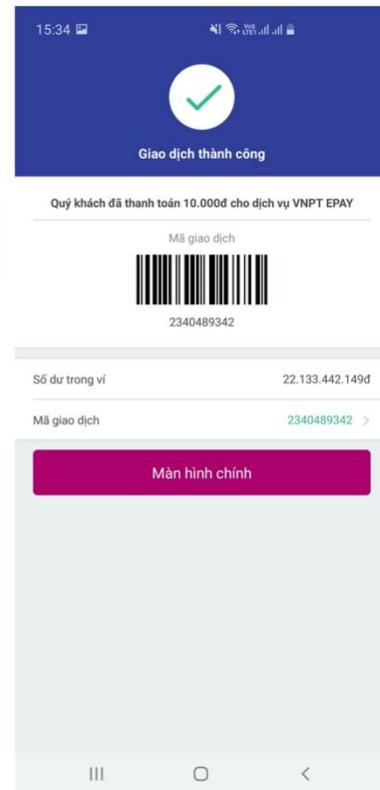
Step 1: Transaction information after scanning the QR code



Step 2: Use password or fingerprint to confirm the transaction



Step 3: The screen informs the successful transaction

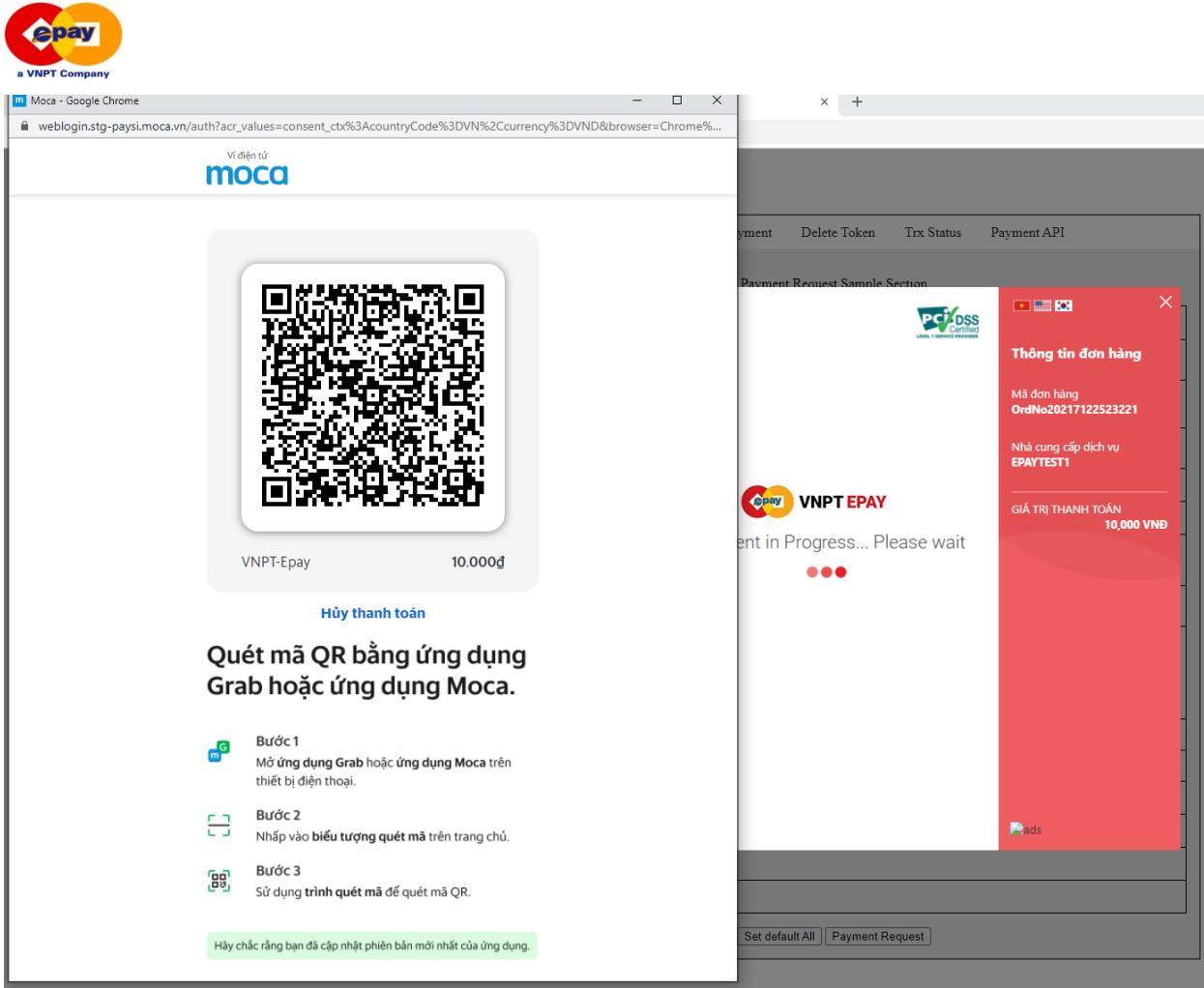




19 Payment guide with Moca

19.1 Payment window

The image shows a payment interface for VNPT EPAY. On the left, there's a sidebar with payment method icons: Thẻ ATM (napas), Thẻ Tín Dụng (VISA, Mastercard, American Express), ZaloPay, MOMO, Moca trên Grab (highlighted with a red border), Chuyển khoản, and Trả góp. A search bar at the top right contains the text "moca". In the center, a red overlay displays "Thông tin đơn hàng" (Order information) with fields for "Mã đơn hàng" (OrdNo20217121224896) and "Nhà cung cấp dịch vụ" (EPAYTEST1). At the bottom right of the red overlay, it says "GIÁ TRỊ THANH TOÁN" (Value) and "10,000 VNĐ". The background of the main interface is white.



19.2 Download Moca app in Sandbox version

Download Grab app (Sandbox version) for android only here

https://drive.google.com/file/d/1wMyKKFmHFYSPymnj_JyJzORPIFEt0NQU/view?usp=sharing

Please send the phone number to the team to activate the login account.

After downloading the app:

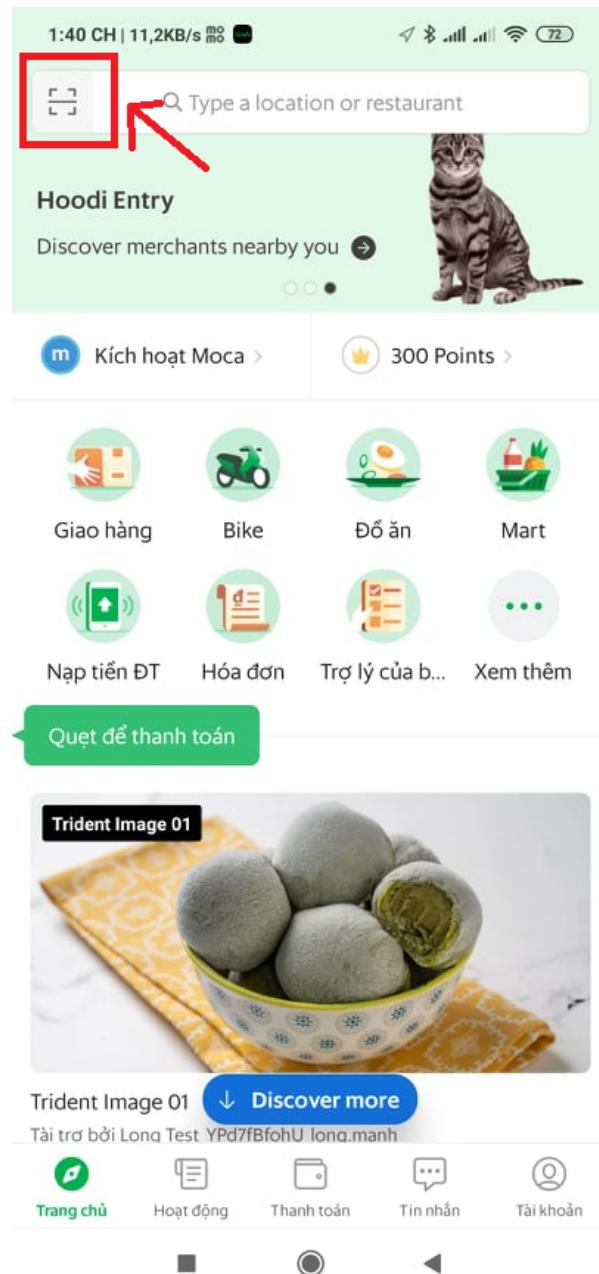
- Implement the application installation on the phone.
- Start the application, log in the account with the phone number sent.

Login to Moca wallet.



19.3 Use Moca app (Sandbox version) & scan QR to pay

Step 1: Login to Moca e-wallet and tap “Scan QR” icon:



Step 2: Access mobile phone camera to scan the QR code in the payment screen (screen no.2 at section 16.1) then follow payment verification steps as requested by Moca e-wallet to complete payment.



a VNPT Company

13:32 4G 1° •

13:32 Vol 4G 83% ■

Ví điện tử
moca

Xin chào, Qa.

Vui lòng xác nhận các thông tin sau để hoàn tất thanh toán

VNPT-Epay

Thanh toán

Tổng cộng

10.000 ₫



Thanh toán qua

Moca wallet

2.994.000 ₫

Để hoàn tất thanh toán, vui lòng quay về lại VNPT-Epay

Xác nhận

Hủy

Bằng cách tiếp tục, bạn đã đọc và đồng ý với [Điều khoản và điều kiện của Moca](#).

Quay lại trang thanh toán của cửa hàng để tiếp tục hoặc giao dịch sẽ thất bại.

OK

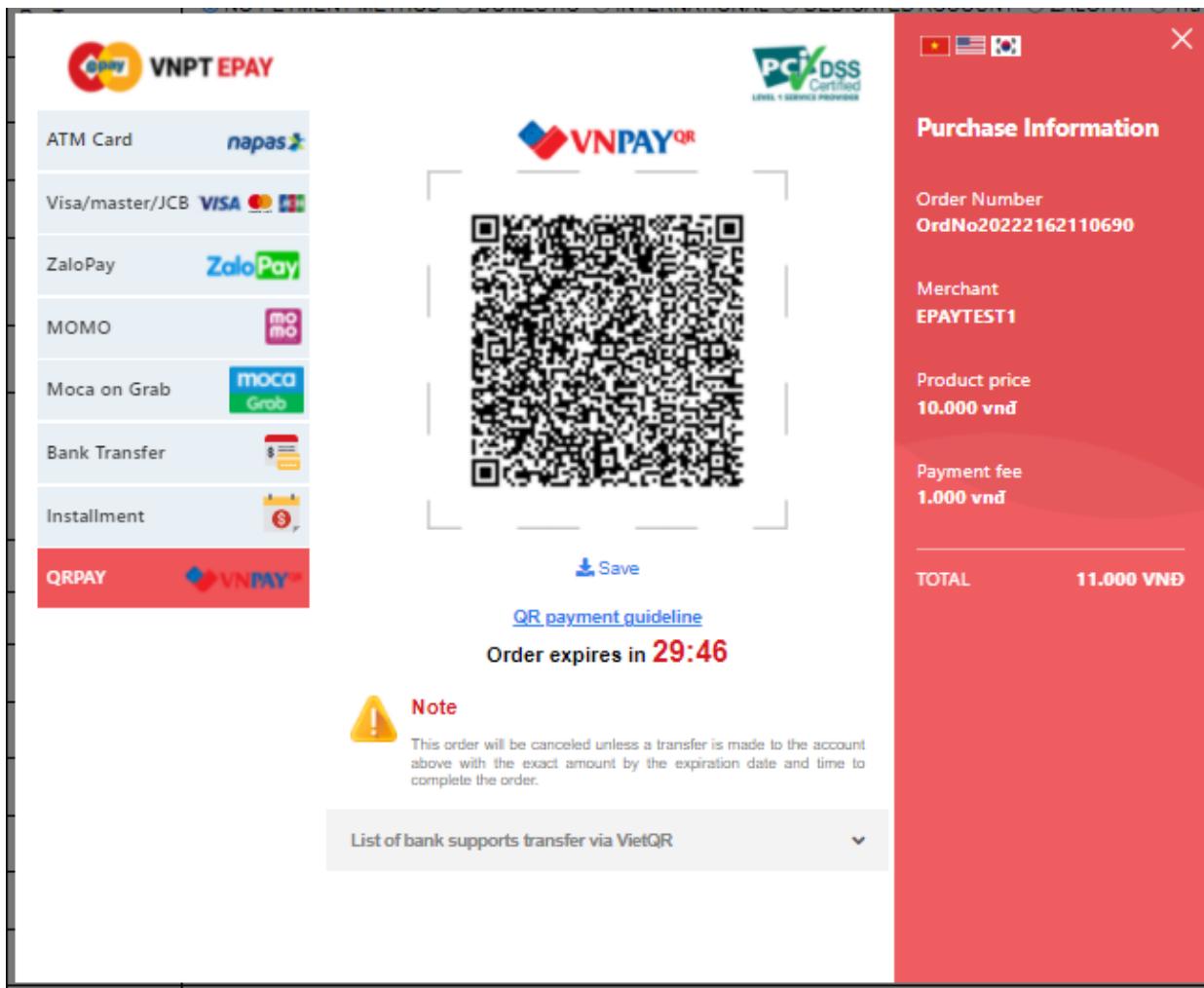


20 Payment guide with VNPAYQR

20.1 Payment window

The screenshot shows a payment interface for VNPT EPAY. On the left, a sidebar lists various payment methods: ATM Card (napas), Visa/master/JCB (VISA, MasterCard), ZaloPay, MOMO, Moca on Grab (moca, Grab), Bank Transfer, Installment, and QR PAY. The QR PAY option is highlighted with a red background. In the center, there is a large button labeled "VNPAYQR". To the right, a "Purchase Information" panel displays the following details:

Purchase Information	
Order Number	OrdNo20222162110690
Merchant	EPAYTEST1
Product price	10.000 vnd
Payment fee	1.000 vnd
TOTAL	11.000 VND



20.2 Download MSB mBank app in Sandbox version

Download MSB mBank application (Sandbox version) and contact the integration support staff

- After downloading the app:
- Install the application on the phone (android).
- Start the application, login account with information:
 - + User: 0984470410, 0332032001, 0332032002, 0332032003
 - + Pass 111111
 - + Otp 11111111
 - + Input PIN: User-configured authentication

Login to MSB mBank app

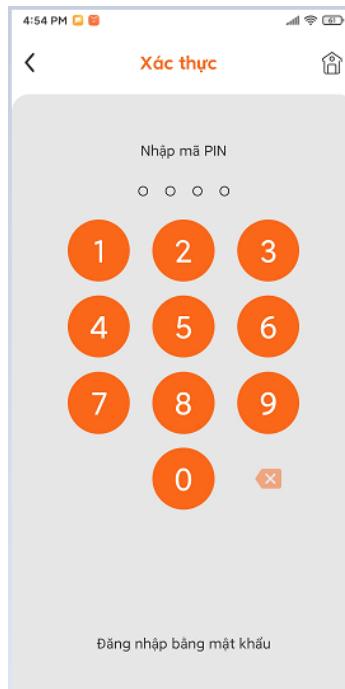
20.3 Use MSB mBank app (Sandbox version) & scan QR to pay



Step 1: Login to MSB mBank and open the QR code scanning feature



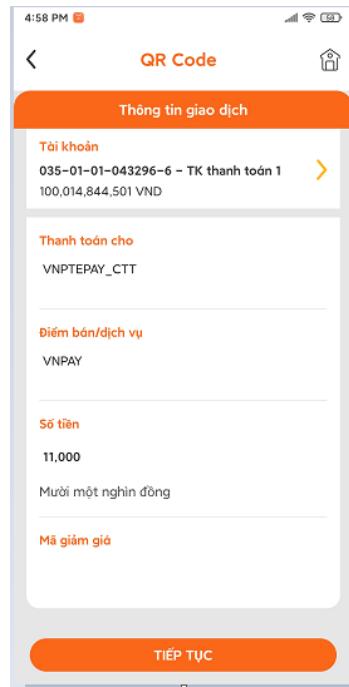
Step 2: Input the preconfigured confirmation PIN





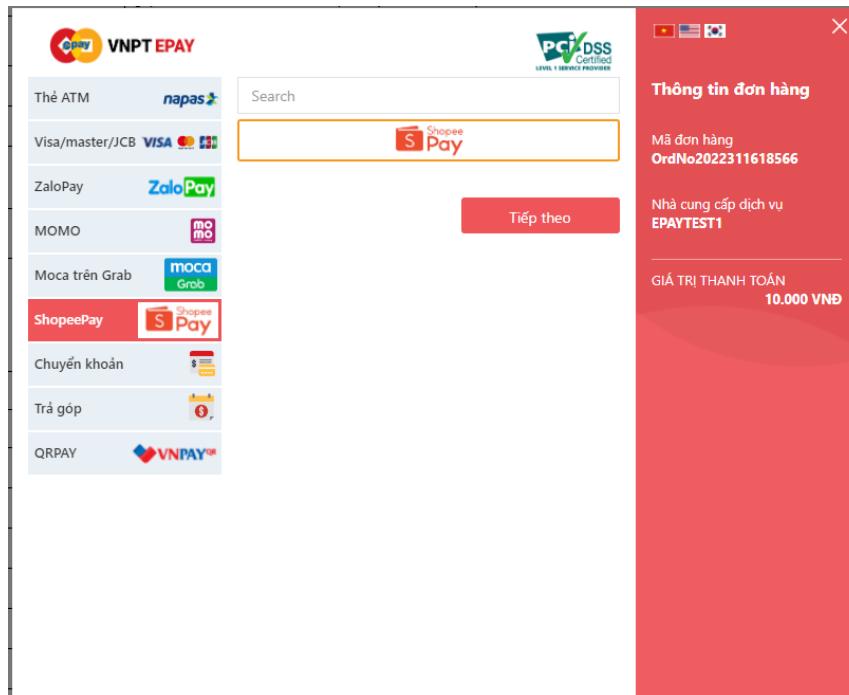
Step 3: Bring up the phone camera to scan the QR code displayed on the payment screen.

After that, complete the payment verification steps as required by MSB mBank to complete the payment.



21 Payment guide with ShopeePay

21.1 Payment window





21.2 Download Shopeepay app (sandbox version)

Download Shopeepay app (Sandbox version) using the link below:

Android: <https://i.diawi.com/PsnFRD>

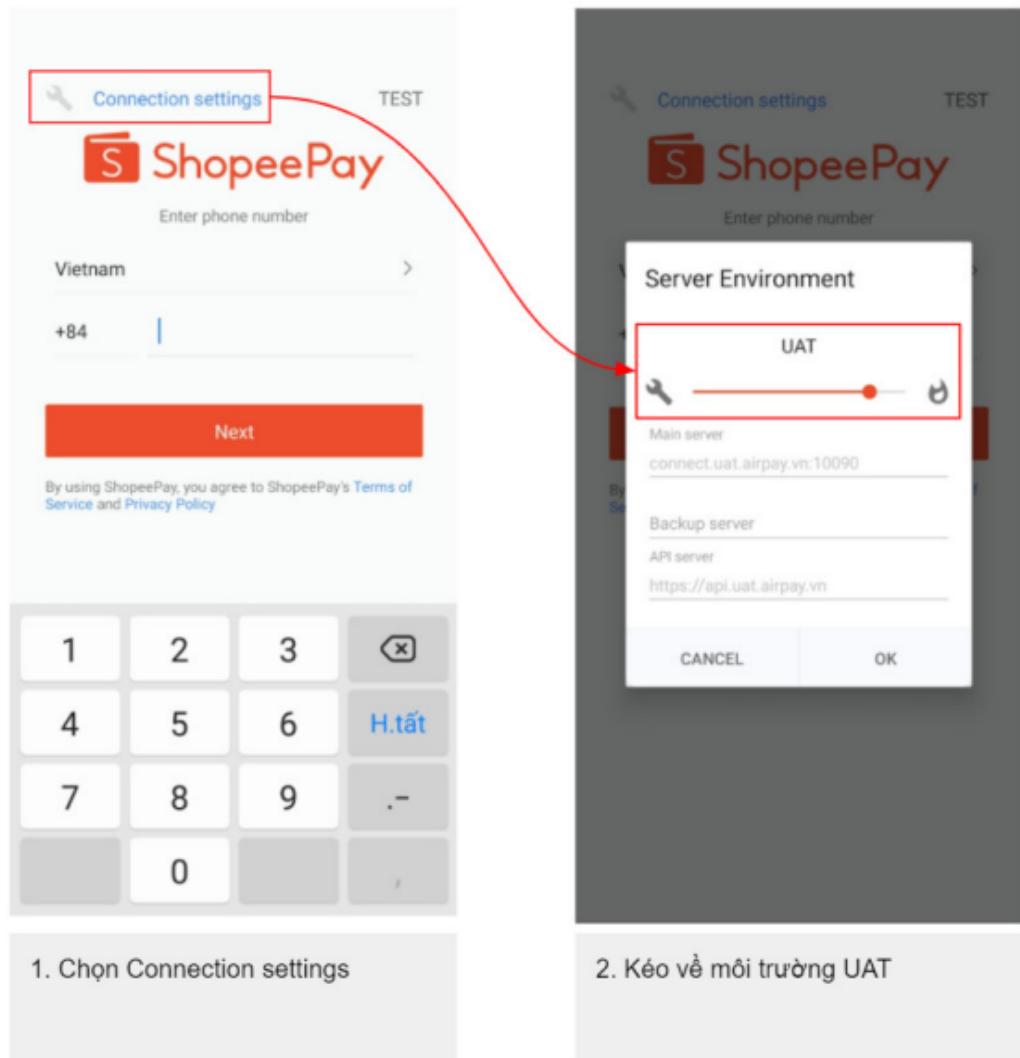
iOS: <https://i.diawi.com/jxL7Mx>

After the download is completed:

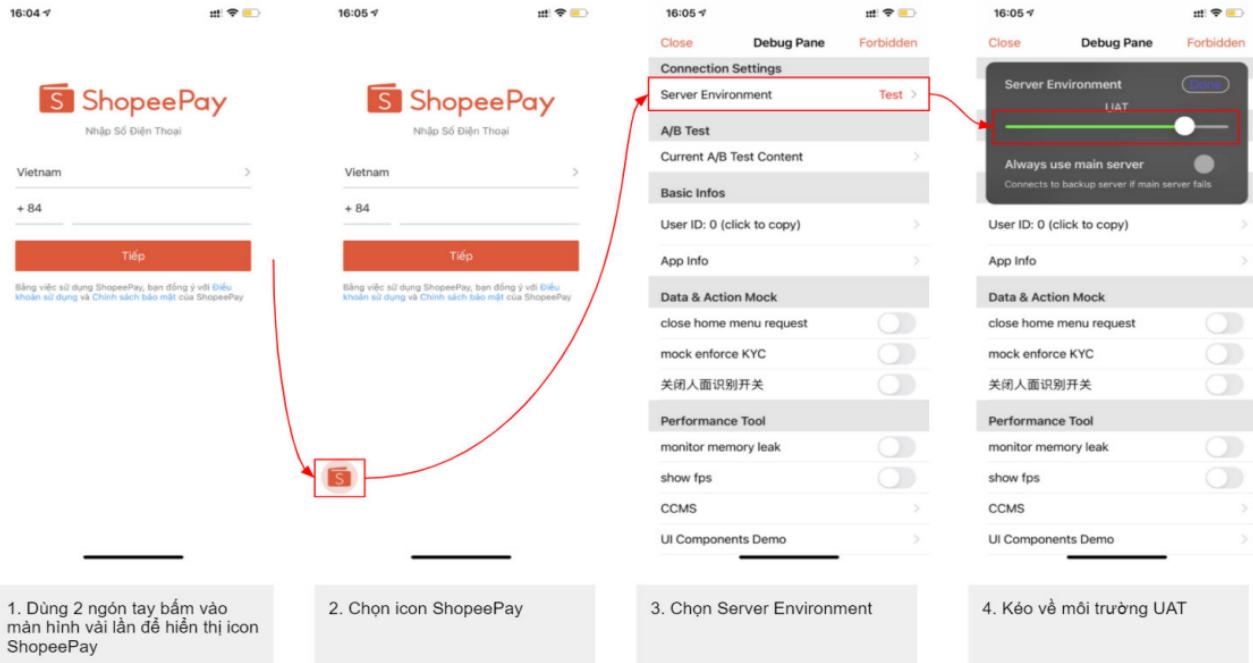
- Install the app in mobile.
- Run and login the app using one of these mobile number below: 0971111111, 0922777777, 0933888888, 0933777777, 0933666666
- Password: 135790
- Authentication code (OTP): 111111

After logging in, you need to follow the instructions below to get into the UAT environment:

- With Android

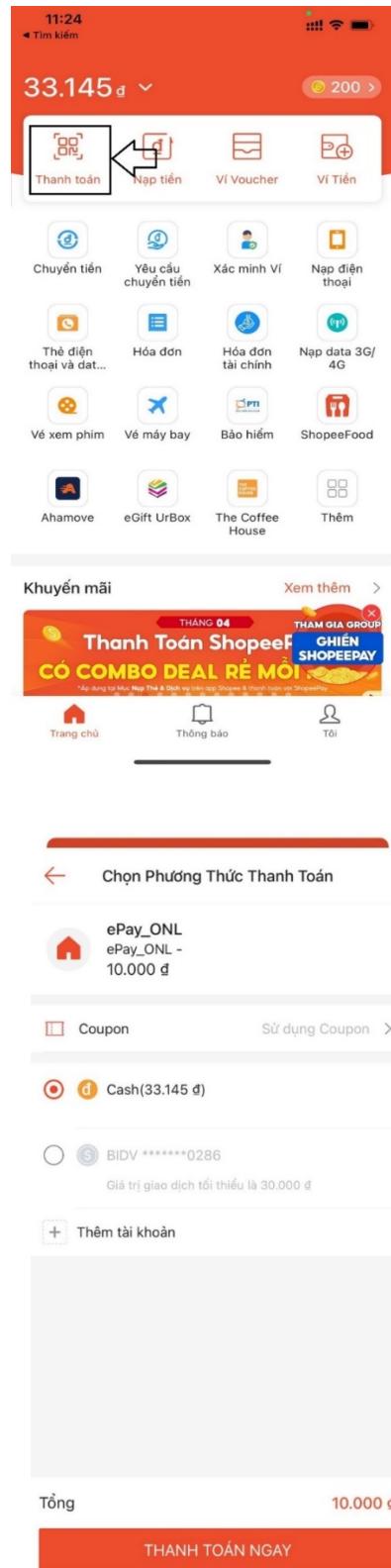


- +) Step 1: Select Connection settings
- +) Step 2: Pull back to UAT environment
- With iOS:



- + Step 1: Use 2 fingers to tap the screen a few times to display the Shopeepay icon
- + Step 2: Select Shopeepay icon
- + Step 3: Select server Environment
- + Step 4: Pull back to UAT environment

21.3 Use ShopeePay app (Sandbox version) payment with PC



Step 1: Login to ShopeePay and open the QR code scanning feature:

Step 2: Bring up the camera to perform QR code scanning.

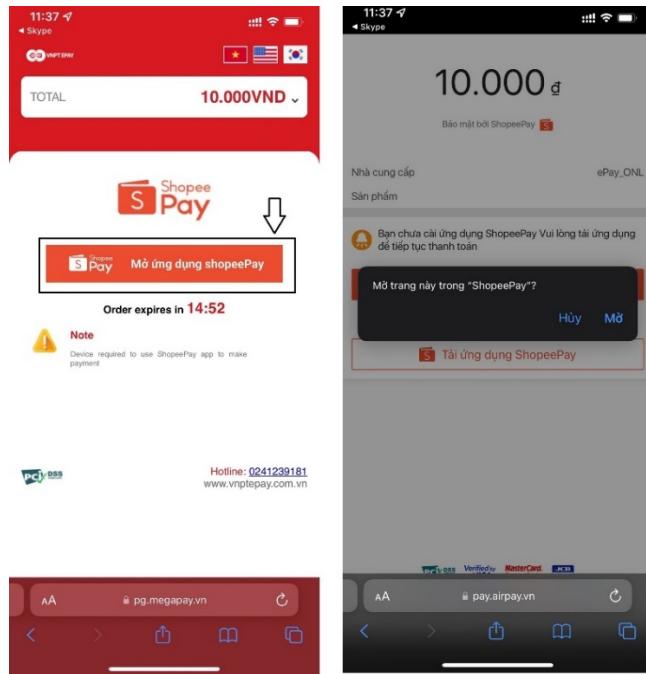
Then perform the payment verification steps required by Shopeepay wallet to complete the payment:



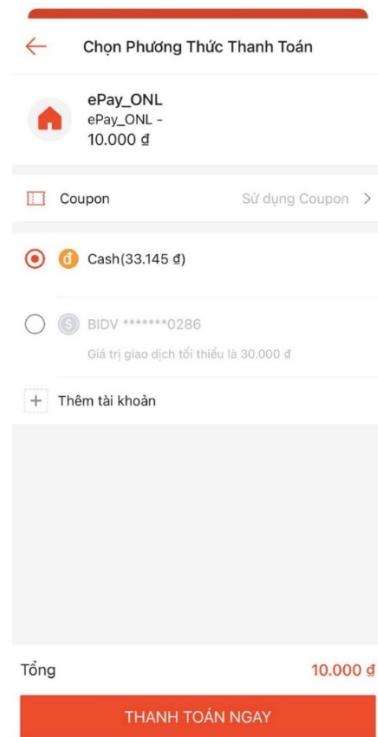
21.4 Use ShopeePay app (Sandbox version) payment with Mobile

Step 1: After choosing to pay with ShopeePay in MGP will show this screen.

At this screen click on “Open ShopeePay app” and agree to open shopeePay app



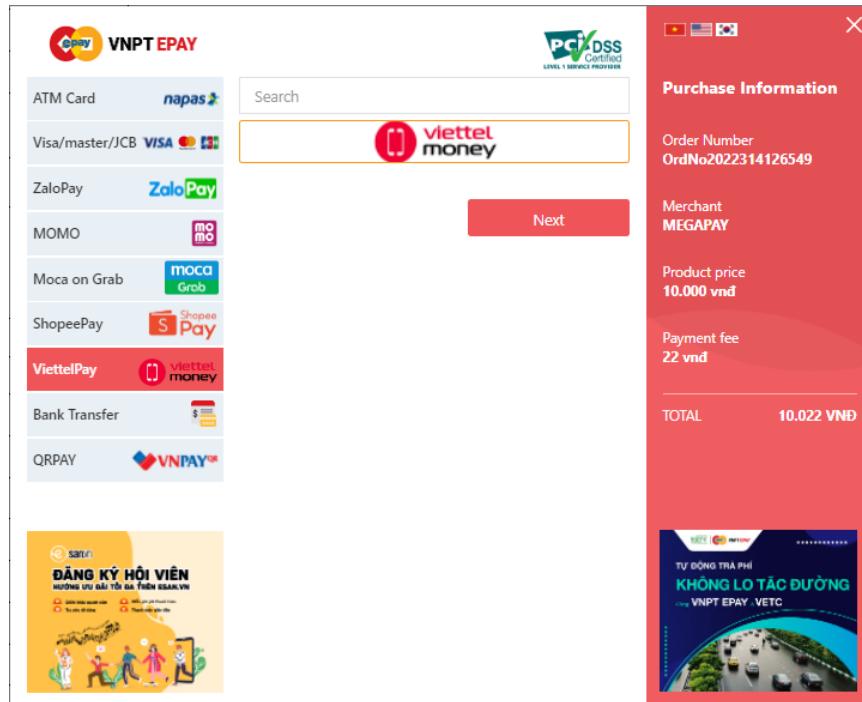
Step 2: Then perform the payment verification steps required by Shopeepay wallet to complete the payment:



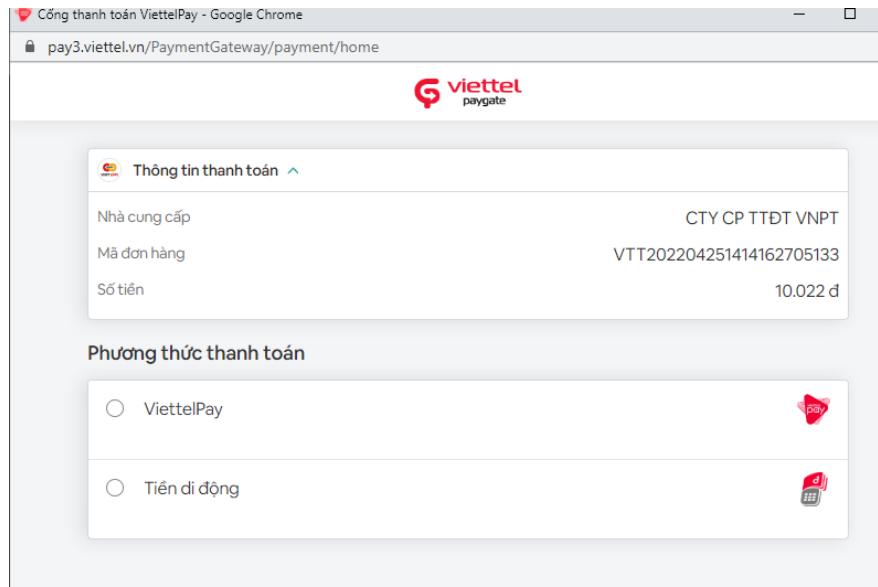


22 Payment guide with ViettelPay

22.1 Payment window



Click button ‘Next’





- Choose the source of money to pay
 - Then enter the payment account
- For example in a sandbox environment use the following account:

User	Pass
0983725525	111111

pay3.viettel.vn/PaymentGateway/payment/home

Nhà cung cấp CTY CP TTĐT VNPT
Mã đơn hàng VTT202204251414162705133
Số tiền 10.022 đ

Phương thức thanh toán

ViettelPay

Vui lòng đăng nhập bằng tài khoản Viettel Money để tiếp tục thanh toán.

Số điện thoại _____

Mật khẩu _____

Đăng nhập

Tiền di động