



# **OUTBOUND Payment Document**

Merchant Development Department

## Version Control

Document Statues	Document Name	Outbound Payment	Confidentiality level	Restricted
<input checked="" type="checkbox"/> Draft	Author	Ni Shenzhou	Completion date	June 5, 2022
<input type="checkbox"/> Revised Version	Reviewed by		Last review date	
<input type="checkbox"/> Officially released	Approved by	Wenbo Tang	Last approval date	

## Revision History

Versions	Revision Date	Revised content	Revisionist
1.3	2022/07/25	<ol style="list-style-type: none"> <li>1. Add applyStatus field to all responses</li> <li>2. Add applyStatus value list</li> <li>3. Remove 100-200 value from responseCode list</li> <li>4. Add deviceId field(Required) to all responses(except notification response) and notification request .</li> <li>5. Add deviceId field(Optional) to all request(except notification request).</li> </ol>	Yue Shen
1.2.5	2022/07/22	<ol style="list-style-type: none"> <li>6. Add the originalMisOrderNo field to the query order response</li> <li>7. Add cardOrganization available value list</li> <li>8. Add error code 114</li> </ol>	Yue Shen
1.2.4	2022/07/18	<ol style="list-style-type: none"> <li>9. Remove request isEmvTag parameter, response will contain emvTag when the order has emvTag</li> <li>10. Change emvTag field name Aid to aid</li> </ol>	Yue Shen
1.2.3	2022/07/15	<ol style="list-style-type: none"> <li>11. Add responseCode, 112,113,201,202</li> <li>12. Correct notification variable name, swapped requestTime and responseTime .</li> </ol>	Yue Shen
1.2.2	2022/07/11	<ol style="list-style-type: none"> <li>13. Improve document content</li> </ol>	Yue Shen
1.2.1	2022/07/11	<ol style="list-style-type: none"> <li>14. Change the parameter tipsAmount to tipAmount</li> <li>15. The currency of all amounts is AED</li> </ol>	Yue Shen
1.2	2022/07/10	<ol style="list-style-type: none"> <li>16. Change chapter's name</li> <li>17. Make MisOrderNo uniqueness</li> <li>18. Add responseCode value definition (201,202)</li> <li>19. Update Sale and Void interaction diagram</li> <li>20. Add merchantOrderNo parameter</li> </ol>	Yue Shen
1.1.4	2022/07/05	Add connection heartbeat detection api	Ni Shenzhou
1.1.3	2022/06/01	Improve document structure	Yue Shen
1.1.2	2022/06/01	Bank card transactions add TC and card type fields return	Ni Shenzhou
1.1.1	2022/05/18	Add payment transaction interface	Ni Shenzhou

# Contents

<b>1. INTRODUCTION.....</b>	<b>4</b>
<b>2. TERMS AND DEFINITIONS.....</b>	<b>5</b>
messageType .....	5
requestTime .....	5
messageId .....	5
functionName .....	5
applyStatus.....	5
amount .....	5
misOrderNo .....	5
payMethod .....	5
printReceiptType .....	5
EMVTag .....	5
voucherNo.....	6
referenceNo .....	6
authorizationNo.....	6
cardType .....	6
responseTime .....	6
merchantOrderNo .....	6
responseCode .....	6
<b>3. FAST INTEGRATION .....</b>	<b>8</b>
3.1. RESOURCE INTRODUCTION .....	8
3.2. QUICK CALL EXAMPLE .....	8
<b>4. TRADING.....</b>	<b>10</b>
4.1. SALE .....	10
4.2. VOID .....	13
4.3. QUERYING ORDERS .....	16
4.4. SALE TRANSACTION NOTIFICATION .....	18
4.5. SEND HEARTBEAT .....	21
<b>5. APPENDIX .....</b>	<b>23</b>
5.1. APPLY STATUS CODE .....	23

# 1.Introduction

## 2. Terms and Definitions

### messageType

Message types. Divided into **Request** and **Response** messages, each request should have a response, and they are associated through messageId.

messageType value
Request
Response

### requestTime

unix epoch time milliseconds

### messageId

The messageId of the request and the messageId of the response are identical and are used to match the request and the response. The recipient of the request should copy the messageId of the request into the response as-is.

### functionName

Used for function routing, the receiver classifies the requests according to functionName.

### applyStatus

When the client receives the response of the ecr, it should check the applyStatus field first. If applyStatus = 0, it means that the request was processed correctly, and then read other fields. When applyStatus !=0, it means that this request has failed, and you don't need to read other fields, because the values of other fields are meaningless at this time.

### amount

The currency of all amounts is AED by default

### misOrderNo

mis order number, is unique within a device, the same order number can be sent repeatedly.

### payMethod

Payment methods. The available options are.

messageType value	Description
PAYBY	PAYBY Wallet Payment
BOTIM	BOTIM Wallet Payment
BANK_CARD	Bank Card Payment

### printReceiptType

Receipt to be printed

messageType value	Description
0	no receipts printed (default)
1	Print customer receipt
2	Print merchant receipt
3	Print customer receipts and merchant receipt

### EMVTag

Related data is generated by the EVM kernel during bank card transactions.

Field Name	Type	Description
aid	String	Application Identifier
atc	String	Application Transaction Counter
tsi	String	Transaction Status Information
tvr	String	Terminal verification results
cid	String	Cryptographic information data
arqc	String	Application Cipher AC
arpc	String	
tc	String	
appName	String	Application Name
appLabel	String	Application Tags

### **voucherNo**

Streaming number, server or POSP system tracking number. Generated by the terminal and sent to the server, usually placed in the ISO8583 11 domain.

### **referenceNo**

Reference number RRN. generated by the payment system and returned to the terminal, usually placed in the ISO 8583 37 field.

### **authorizationNo**

Authorization number. Generated by the payment system and returned to the terminal, usually placed in the ISO 8583 38 field.

### **cardType**

The type of card used for bank card transactions.

messageType value	Description
Magnetic Stripe	Magnetic Stripe Card Transaction
Contactless	card waving/contactless transaction
Contact	Card Insert/Contact Card Transaction
Manual	Manual card number entry transaction

### **cardOrganization**

cardOrganization
VISA
MASTERCARD

### **responseTime**

unix epoch time milliseconds

### **merchantOrderNo**

Merchant order number, the order number generated by the pos machine, which is unique within a merchant system

### **responseCode**

Value	Description
-------	-------------

<b>00</b>	Order is success
<b>201</b>	Order is processing
<b>202</b>	Order is failure
<b>01-99</b>	Order is failure, value is the reason of the failure

## 3. Fast Integration

### 3.1. Resource Introduction

Copy the obtained aar file to the libs folder of the corresponding module (if there is no such directory, create a new one), and then add the following code to the build.gradle file.

```
android {
    repositories {
        flatDir { dirs 'libs' }
    }
}

dependencies {
    implementation (name: 'payby-export-payment-xxx', ext: 'aar')
}
```

After the import of aar is complete, rebuild the project.

### 3.2. Quick call example

```
fun startTransaction() {
    val request = Request()
    request.messageId = System.currentTimeMillis().toString()
    request.messageType = "Request"
    request.requestTime = System.currentTimeMillis()
    request.functionName = "SALE"
    request.misOrderNo = BuildConfig.APPLICATION_ID + "_" + System.currentTimeMillis().toString()
    request.totalAmount = 12.12
    request.payMethod = "BANK_CARD"
    request.subject = "a bottle of beer"
    request.printReceiptType = 1
    val jsonString = Gson().toJson(request)
    PayByExportPaymentService.getInstance().startTransaction(jsonString, responseCallback)
}

private val responseCallback = object : ExportResponseCallback.Stub() {

    override fun onResponse(response: String ? ) {
        Logger.e(App.TAG, msg: "response:$response")
        ResultActivity.startAction(baseContext, response)
        finish()
    }
}
```

For more usage examples, please refer to the demo

#### !!! Attention

When a Request type message is received, a Response type message needs to be sent to the PayBy payment application.



## I Sample request

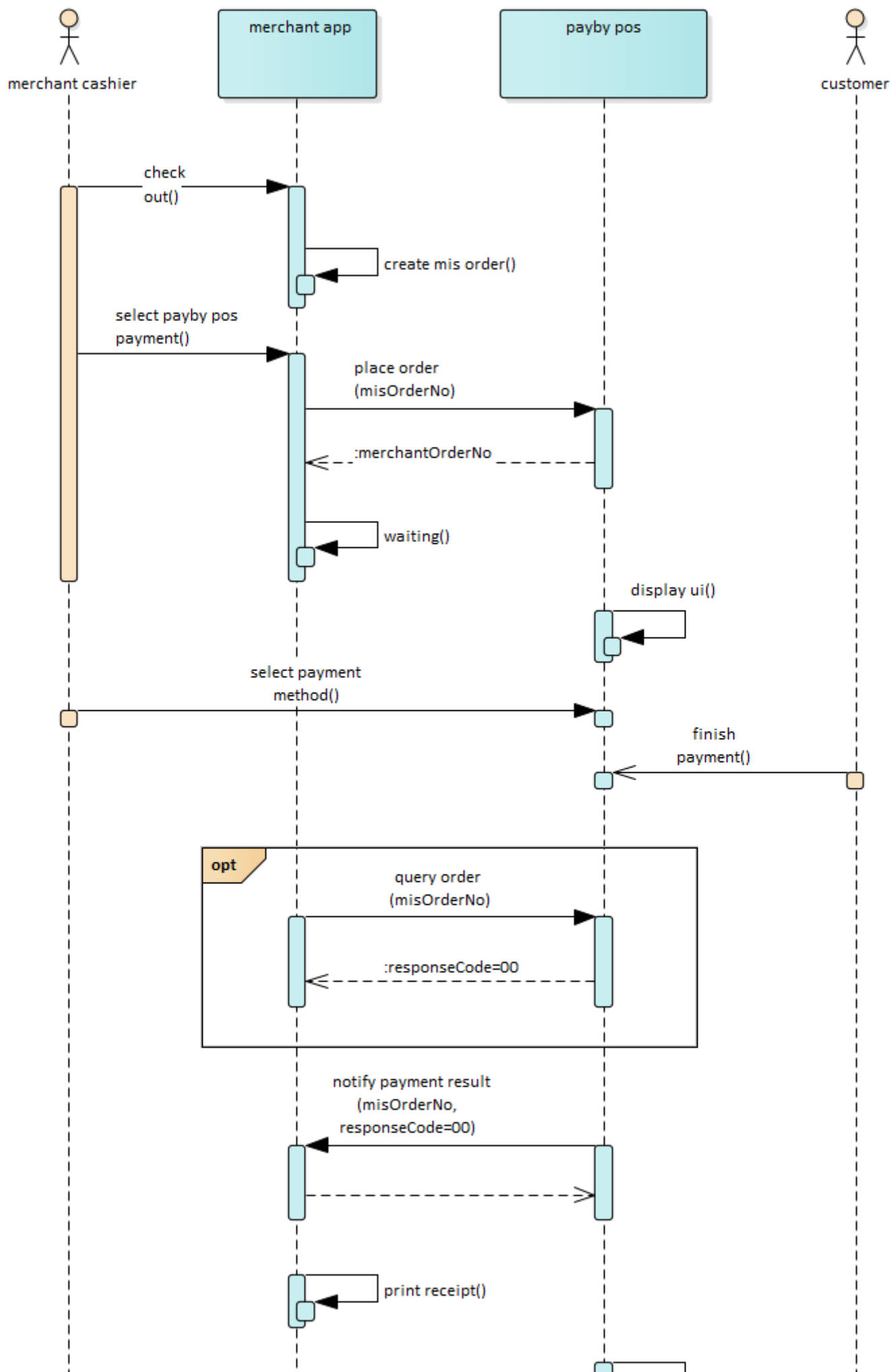
```
{  
  "messageType": "Request",  
  "messageId": "123456",  
  "functionName": "SALE",  
  "requestTime": 1581404947666,  
  "misOrderNo": "M965739182419",  
  "subject": "Your subject",  
}
```

## 4. Trading

### 4.1. Sale

The merchant app initiates a sale request to payby pos, the pos machine responds immediately and opens ui, accepts the user's swipe card, then submits the online transaction and receives the connection transaction result and notifies the merchant app. if the merchant app does not receive a response, it can initiate the query result itself.

# sd payby-pos-purchase-sequence-en



## | Request Parameters

Field Name	Type	Is mandatory	Description
requestTime	Number	M	Request Time
messageType	String	M	Message type. Request
functionName	String	M	The name of the requested API. SALE
messageld	String	M	Unique identification of the message
misOrderNo	String	M	mis Order number
subject	String	M	Product Name
totalAmount	Number	M	The total amount of the order (in dollars). Example: 12.00
tipAmount	Number	O	Tip amount
payMethod	String	O	Payment Method
printReceiptType	Number	O	Print Receipt Method.
deviceId	Number	O	device number assigned by payby

## | Sample request

```
{
  "messageType": "Request",
  "functionName": "SALE",
  "requestTime": 1581404947666,
  "messageld": "KX121313124144141",
  "misOrderNo": "M965739182419",
  "subject": "Your subject",
  "totalAmount": 12.00,
  "tipAmount": 1.00,
  "payMethod": "BANK_CARD",
  "printReceiptType": 3,
  "deviceId": "123555"
}
```

## | Response parameters

Field Name	Type	Is mandatory	Description
applyStatus	Number	M	Check applyStatus first
responseTime	Number	M	Response time, unix epoch time milliseconds.
responseCode	String	M	Response Code
responseMessage	String	O	Response Description
messageType	String	M	Message type. response
functionName	String	M	The name of the requested API. SALE
messageld	String	M	Unique identification of the message
misOrderNo	String	M	mis order number
subject	String	O	Product Name
totalAmount	Number	M	Total order amount
tipAmount	Number	O	Spending tip amount
payMethod	String	O	Payment Method
appVersion	String	M	Application version number
packageName	String	M	Application Package Name
merchantName	String	O	Merchant Name
merchantNo	String	O	Merchant Number

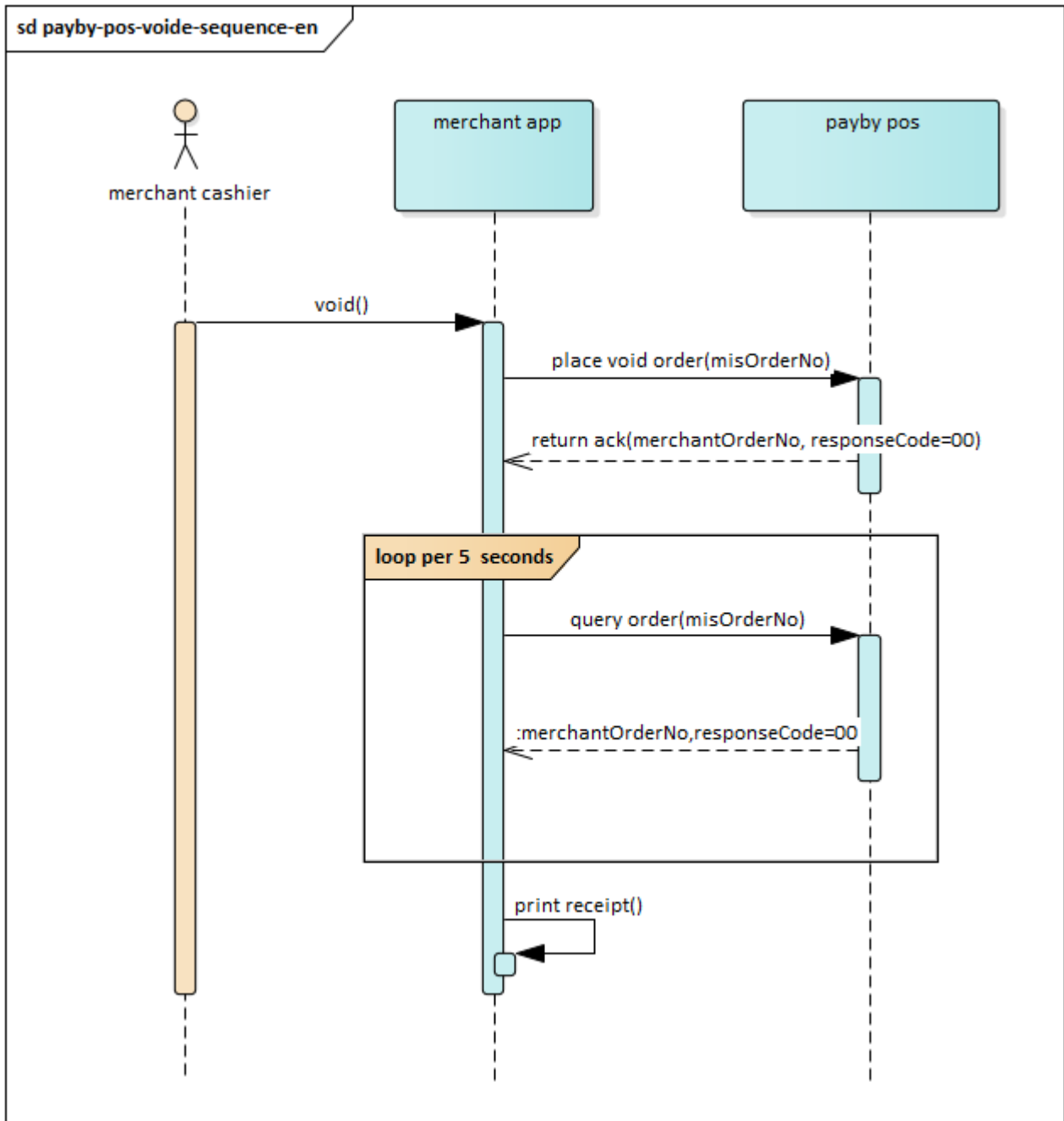
<b>merchantOrderNo</b>	String	M	Merchant order number, generated by the pos machine, which is unique within a merchant system
<b>terminalNo</b>	String	O	Terminal number
<b>deviceId</b>	Number	M	device number assigned by payby

## I Sample response

```
{
  "applyStatus": 0,
  "responseTime": 1581404947666,
  "responseCode": "201",
  "responseMessage": "Success",
  "messageType": "Response",
  "functionName": "SALE",
  "messageId": "KX121313124144141",
  "misOrderNo": "M965739182419",
  "subject": "Your subject",
  "totalAmount": 12.00,
  "tipAmount": 1.00,
  "payMethod": "BANK_CARD",
  "appVersion": "1.2.121",
  "packageName": "com.payby.pos",
  "merchantName": "Shanghai Tavern",
  "merchantNo": "5131631641641",
  "merchantOrderNo": "05c97402-8f80-4423-af9d-53cd675a3a54",
  "terminalNo": "1453151",
  "deviceId": "123555"
}
```

## 4.2. Void

Merchant app initiates VOID request to payby pos, pos machine submits online transaction and receives connection transaction result and returns to merchant app synchronously. if merchant app does not receive response, it can initiate query result by itself.



### | Request Parameters

Field Name	Type	Is mandatory	Description
requestTime	Number	M	
messageType	String	M	Request
functionName	String	M	VOID
messageld	String	M	
misOrderNo	String	M	
originalMisOrderNo	String	M	Original transaction mis order number
printReceiptType	Number	O	
deviceId	Number	O	device number assigned by payby

### | Sample request

```
{
  "messageType": "Request",
  "functionName": "VOID",
  "requestTime": 1581404947666,
  "messageId": "KX121313124144141",
  "misOrderNo": "M965739182419",
  "originalMisOrderNo": "M965739182419",
  "printReceiptType": 3,
  "deviceId": "123555"
}
```

## Response parameters

Field Name	Type	Is mandatory	Description
applyStatus	Number	M	Check applyStatus first
responseTime	Number	M	Response time, unix epoch time milliseconds.
responseCode	String	M	Response Code
responseMessage	String	O	Response Description
messageType	String	M	Message type. response
functionName	String	M	The name of the requested API. VOID
messageId	String	M	Unique identification of the message
misOrderNo	String	M	mis order number
originalMisOrderNo	String	M	Original mis order number
subject	String	O	Product Name
totalAmount	Number	M	Total order amount
tipAmount	Number	O	Spending amount
payMethod	String	O	Payment Method
appVersion	String	M	Application version number
packageName	String	M	Application Package Name
merchantName	String	O	Merchant Name
merchantNo	String	O	Merchant Number
merchantOrderNo	String	M	This is merchantOrderNo used by void.
terminalNo	String	O	Terminal number
deviceId	Number	M	device number assigned by payby
emvTag	EmvTag	O	Some of the EMV data in the transaction. Reference <a href="#">EmvTag</a>

## Sample response

```
{
  "applyStatus": 0,
  "responseTime": 1581404947666,
  "responseCode": "201",
  "responseMessage": "Success",
  "messageType": "Response",
  "functionName": "VOID",
  "messageId": "KX121313124144141",
  "misOrderNo": "M965739182419",
  "originalMisOrderNo": "M965739182418",
  "subject": "Your subject",
}
```

```

"totalAmount": 12.00,
"tipAmount": 1.00,
"payMethod": "BANK_CARD",
"appVersion": "1.2.121",
"packageName": "com.payby.pos",
"merchantName": "Shanghai Tavern",
"merchantNo": "5131631641641",
"merchantOrderNo": "05c97402-8f80-4423-af9d-53cd675a3a54",
"terminalNo": "1453151",
"deviceId": "123555",
"emvTag": {
  "aid": "A00000000333",
  "atc": "892311",
  "tsi": "8000",
  "tvr": "0000000000",
  "cid": "921303",
  "tc": "012131314142141231",
  "arqc": "983112345678",
  "arpc": "983112345678",
  "appName": "Visa Debit",
  "appLabel": "Visa"
}
}

```

### 4.3. Querying orders

This interface provides the query of all orders. Merchants can actively query the order status through the query order interface to complete the next step of business logic. Cases where the query interface needs to be called.

1. When there is an abnormality in the merchant's backend, network, server, etc. The merchant's system eventually does not receive the payment notification.
2. After calling the payment interface, a system error or unknown transaction status situation is returned.

#### | Request Parameters

Field Name	Type	Is mandatory	Description
requestTime	Number	M	
messageType	String	M	Request
functionName	String	M	GET_ORDER
messageld	String	M	
misOrderNo	String	M	
transactionType	String	M	SALE,VOID
deviceId	Number	O	device number assigned by payby

#### | Sample request

```

{
  "messageType": "Request",
  "functionName": "GET_ORDER",
  "transactionType": "SALE",

```



```

"requestTime": 1581404947666,
"messageId": "KX121313124144141",
"misOrderNo": "M965739182419",
"deviceId": "123555"

```

```

}
```

### Response parameters

Field Name	Type	Is mandatory	Description
applyStatus	Number	M	Check applyStatus first
responseTime	Number	M	Response time
responseCode	String	M	Response Code
responseMessage	String	M	Response Description
messageType	String	M	Response
functionName	String	M	GET_ORDER
transactionType	String	M	Transaction type. SALE
messageId	String	M	Unique identification of the message
misOrderNo	String	M	mis order number
originalMisOrderNo	String	O	When the transactionType is SALE, this field is empty. When the transactionType is VOID, this field is mandatory and is to indicate the order to be voided.
orderNo	String	M	PayBy System Order Number
subject	String	M	Product Name
totalAmount	Number	M	Total order amount
tipAmount	Number	O	Spending amount
payMethod	String	O	Payment Method
appVersion	String	M	Application version number
packageName	String	M	Application Package Name
merchantName	String	M	Merchant Name
merchantNo	String	M	Merchant Number
merchantOrderNo	String	M	Merchant order number, generated by the pos machine, which is unique within a merchant system
terminalNo	String	O	Terminal number
deviceId	Number	M	device number assigned by payby
cardType	String	O	Card type. For example: Contact
cardOrganization	String	O	card organizations. For example: VISA
cardNoMask	String	O	Card number mask. For example: 123456 ****0888
cardHolderName	String	O	Cardholder Name
voucherNo	String	O	System trace audit number (STAN)
referenceNo	String	O	Reference number
authorizationNo	String	O	Authorization Number
transactionDate	String	O	Transaction Date
transactionTime	String	O	Trading Hours
emvTag	EmvTag	O	Some of the EMV data in the transaction. Reference <a href="#">EmvTag</a>

### Sample response

```

{
  "applyStatus": 0,
  "responseTime": 1581404947666,
  "responseCode": "00",
  "responseMessage": "Success",
  "messageType": "Response",
  "functionName": "GET_ORDER",
  "transactionType": "SALE",
  "messageId": "KX121313124144141",
  "misOrderNo": "M965739182419",
  "originalMisOrderNo": "M965739182418",
  "orderNo": "1234567",
  "subject": "Your subject",
  "totalAmount": 12,
  "tipAmount": 1,
  "payMethod": "BANK_CARD",
  "appVersion": "1.2.121",
  "packageName": "com.payby.pos",
  "merchantName": "Shanghai Tavern",
  "merchantOrderNo": "05c97402-8f80-4423-af9d-53cd675a3a54",
  "merchantNo": "5131631641641",
  "terminalNo": "1453151",
  "deviceId": "123555",
  "cardType": "Contactless",
  "cardOrganization": "VISA",
  "cardNoMask": "123456*****0008",
  "cardHolderName": "AAA BBBS",
  "voucherNo": "123456",
  "transactionTime": "1010",
  "transactionDate": "0510",
  "referenceNo": "123456",
  "authorizationNo": "12345678",
  "emvTag": {
    "aid": "A0000000333",
    "atc": "892311",
    "tsi": "8000",
    "tvr": "0000000000",
    "cid": "921303",
    "tc": "012131314142141231",
    "arqc": "983112345678",
    "arpc": "983112345678",
    "appName": "Visa Debit",
    "appLabel": "Visa"
  }
}

```

## 4.4. Sale transaction notification

The pos will notify the SALE transaction result to the mis, and the mis side needs to send a response message to the pos after receiving the notification. If the pos does not receive the response, a message notification will be sent every 5 seconds and repeated 3 times.

If the mis system handles the notification correctly, it should return applyStatus=0, and pos will not send further notifications. If the mis system cannot handle the notification properly, it should return applyStatus=105 and pos will send the message again (up to 3 times).

## I Notification parameters

Field Name	Type	Is mandatory	Description
requestTime	Number	M	request time
responseCode	String	M	Response Code
responseMessage	String	M	Response Description
messageType	String	M	Request
functionName	String	M	NOTIFY_ORDER
transactionType	String	M	Transaction type. SALE
messageld	String	M	Unique identification of the message
misOrderNo	String	M	mis order number
orderNo	String	M	PayBy System Order Number
subject	String	M	Product Name
totalAmount	Number	M	Total order amount
tipAmount	Number	O	Spending amount
payMethod	String	O	Payment Method
appVersion	String	M	Application version number
packageName	String	M	Application Package Name
merchantName	String	M	Merchant Name
merchantNo	String	M	Merchant Number
merchantOrderNo	String	M	Merchant order number, generated by the pos machine, which is unique within a merchant system
terminalNo	String	O	Terminal number
deviceId	Number	M	device number assigned by payby
cardType	String	O	Card type. For example: Contact
cardOrganization	String	O	card organizations. For example: VISA
cardNoMask	String	O	Card number mask. For example: 123456 ****0888
cardHolderName	String	O	Cardholder Name
voucherNo	String	O	System trace audit number (STAN)
referenceNo	String	O	Reference number
authorizationNo	String	O	Authorization Number
transactionDate	String	O	Transaction Date
transactionTime	String	O	Trading Hours
emvTag	EmvTag	O	Some of the EMV data in the transaction. Reference <a href="#">EmvTag</a>

## I Sample notification

```
{
  "requestTime": 1581404947666,
  "responseCode": "00",
  "responseMessage": "Success",
```

```

    "messageType": "Request",
    "functionName": "NOTIFY_ORDER",
    "transactionType": "SALE",
    "messageId": "KX121313124144141",
    "misOrderNo": "M965739182419",
    "orderNo": "1234567",
    "subject": "Your subject",
    "totalAmount": 12,
    "tipAmount": 1,
    "payMethod": "BANK_CARD",
    "appVersion": "1.2.121",
    "packageName": "com.payby.pos",
    "merchantName": "Shanghai Tavern",
    "merchantNo": "5131631641641",
    "merchantOrderNo": "05c97402-8f80-4423-af9d-53cd675a3a54",
    "terminalNo": "1453151",
    "deviceId": "123555",
    "cardType": "Contactless",
    "cardOrganization": "VISA",
    "cardNoMask": "123456*****0008",
    "cardHolderName": "AAA BBBS",
    "voucherNo": "123456",
    "transactionTime": "1010",
    "transactionDate": "0510",
    "referenceNo": "123456",
    "authorizationNo": "12345678",
    "emvTag": {
      "aid": "A00000000333",
      "atc": "892311",
      "tsi": "8000",
      "tvr": "0000000000",
      "cid": "921303",
      "tc": "012131314142141231",
      "arqc": "983112345678",
      "arpc": "983112345678",
      "appName": "Visa Debit",
      "appLabel": "Visa"
    }
  }
}

```

## I Response parameters

Field Name	Type	Is mandatory	Description
<b>applyStatus</b>	Number	M	<p>If the mis system handles the notification correctly, it should return applyStatus=0, and pos will not send further notifications.</p> <p>If the mis system cannot handle the notification properly, it should return applyStatus=105 and pos will send the message again (up to 3 times).</p>
<b>responseTime</b>	Number	M	

<b>messageType</b>	String	M	Response
<b>functionName</b>	String	M	NOTIFY_ORDER
<b>messageld</b>	String	M	Messageld of the original notification
<b>misOrderNo</b>	String	M	misOrderNo of the notification

### | Sample response

```
{
  "messageType": "Response",
  "functionName": "NOTIFY_ORDER",
  "responseTime": 1581404947666,
  "messageld": "KX121313124144141",
  "misOrderNo": "M965739182419",
  "response": "SUCCESS",
}
```

## 4.5. Send heartbeat

When you use websocket as a connection method, you need to send a heartbeat periodically to check the validity of the connection. When you do not receive the correct heartbeat back, you need to assume that the connection has failed and need to re-establish the connection immediately.

### | Request Parameters

Field Name	Type	Is mandatory	Description
<b>requestTime</b>	Number	M	
<b>messageType</b>	String	M	Request
<b>functionName</b>	String	M	TERMINAL_HEARTBEAT_CHECK
<b>messageld</b>	String	M	
<b>deviceId</b>	Number	O	device number assigned by payby

### | Sample request

```
{
  "messageType": "Request",
  "functionName": "TERMINAL_HEARTBRAT_CHECK",
  "requestTime": 1581404947666,
  "messageld": "KX121313124144141",
  "deviceId": "123555"
}
```

### | Response parameters

Field Name	Type	Is mandatory	Description
<b>applyStatus</b>	Number	M	Check applyStatus first
<b>responseTime</b>	Number	M	
<b>messageType</b>	String	M	Response
<b>functionName</b>	String	M	TERMINAL_HEARTBEAT_CHECK
<b>messageld</b>	String	M	
<b>deviceId</b>	Number	M	device number assigned by payby

### | Sample response

```
{  
  "applyStatus": 0,  
  "messageType": "Response",  
  "functionName": " TERMINAL_HEARTBRAT_CHECK ",  
  "responseTime": 1581404947666,  
  "messageId": "KX121313124144141",  
  "deviceId": "123555"  
}
```

## 5. Appendix

### 5.1. Apply status code

Value	Description
0	The request was processed correctly
101	Parameter error
104	Too frequent requests
105	System Error
106	Service Timeout
107	Risk control verification failure
109	Product not applied
110	Merchant equipment does not exist
111	Device Access Denied
112	Requests with the same order number cannot be sent repeatedly. Use query order function to get the order status
113	The pos machine is processing other session.
114	The pos machine is executing another query request.