

GCCPAY Payment API Document

1. Introduction

System overview: The GCCPAY payment system is used to complete the payment collection of merchants. The main process of docking is that the merchant creates a payment order through the interface, and guides the payer (buyer) to the GCCPAY payment page. After completing the payment, GCCPAY will guide the payer's browser to redirect to the merchant page, GCCPAY will also notify the merchant through the server.

PS: Please review the newest API Document in GCCPAY our official website:

<https://gccpay.com/documents/api-summary>

2. Update History

Time	Version	Originator	Main changes
20221023	V1.0.0	Xiongfei.Zhao	First edition
20221110	V1.0.1	Xiongfei.Zhao	Added [Notification Callback] module
20221116	V1.1.0	Shaopeng.Li	Add iframe docking method
20230302	V1.1.1	Xiongfei.Zhao	Add [tip], [payerInfo] and [payLink] in [Query Payment Order] method
20230316	V1.1.2	Hengjie.Zhang	Add the 'lockPayMethod' parameter in the URL of payment link

Release time & Release content

- Version V1.0.0 was released on October 23, 2022
- Version V1.0.1 was released on November 10, 2022
- Version V1.1.0 was released on November 16, 2022
- Version V1.1.1 was released on March 2, 2023

- Version V1.1.2 was released on March 16, 2023

Post content

V1.0.0

- The first draft of the payment system Open API

V1.0.1

- Added [Notification Callback] module

V1.1.0

- Added iframe docking method
- Added code demo
- Added [Merchant Information Query] method

V1.1.1

- Add [tip], [payerInfo] and [payLink] in [Query Payment Order] method

V1.1.2

- Add the 'lockPayMethod' parameter to the URL of the payment link which can control the payment method displayed on the cashier page.

3. Call Method

3.1 Term Explanation

Term	Illustration
Merchant	The merchant is the seller in the payment scenario, and the payment system is held by the merchant, and the merchant is responsible for the collection business.
Payer	The buyer pays the seller through the payment page in the payment scene.
Payment order	It is created by the seller calling the create order API of the payment gateway by signature and other information.
Transaction	When the payer pays the seller through the payment page, a certain payment behavior may generate multiple payment transactions, but only one will succeed. If one transaction succeeds, others will be automatically invalidated.
2FA(2-step verification string)	Using Google Identity to improve account security, users need to import the app into the mobile phones, and the verification data needs to be stored in the app.

Client	Contains the ID, key and secret used to calculate the signature.

3.2 Service info

Sandbox environment

Service address: https://sandbox.gcc-pay.com/api_v1/

Communication protocol: HTTP

Encoding: UTF-8

3.3 Public parameters

Public request parameters

HTTP's header:

Header name	Type	Is required	Illustration
x-auth-signature	String	Yes	The string of auth signature
x-auth-key	String	Yes	The key of auth
x-auth-timestamp	String	Yes	The request's timestamp, int32, accurate to seconds
x-auth-sign-method	String	Yes	The signature calculation method is currently unified as HmacSHA256
x-auth-sign-version	String	Yes	The signature calculation method version is currently unified as 1

3.4 Authentication mechanism

1. Construct the header information of the request parameters, and all the information in the signature verification is in the header information
2. Signature calculation method
 - a. Arrange a batch of key-value pairs into the format of key=value. value is processed by urlencode;

- b. Then sort the batch of key-value pairs in ascending dictionary order, and then use the & symbolic link;
- c. Then use the secret applied above to use sha256 to calculate the summary information of the spliced string, and the output is base64 encoding.
- d. The key-value pairs are as follows
 - i. `uri` : the requested url without the root part, for example: `/users/100000/orders`
 - ii. `key` : the applied key
 - iii. `timestamp` : timestamp, int32, accurate to seconds
 - iv. `signMethod` : fixed, "HmacSHA256"
 - v. `signVersion` : fixed, "1"
 - vi. `method` :
 - Create order: `merchant.addOrder`
 - Query order: `order.detail`
 - Query merchant information: `merchant.detail`

3.5 Response Result

Successful response

When the HTTP response status code is `2xx`, the call is successful.

Failure response

When the HTTP response status code is `4xx` or `5xx`, the call fails.

Response Sample

```
1 {
2   "code": "notAllowed", ### Error code
3   "message": "No access", ### Error description
4   "data": ### Error detail
5   [
6     "signature error",
7     {
8       "uri": "/merchants/M448726",
9       "key": "zS83UNCPhVTqBxDHACJ30sImZRKA1zQI",
10      "timestamp": 1672991487,
11      "signMethod": "HmacSHA256",
12      "signVersion": "1",
13      "method": "merchant.detail"
14    }
15  ]
}
```

4. APIS

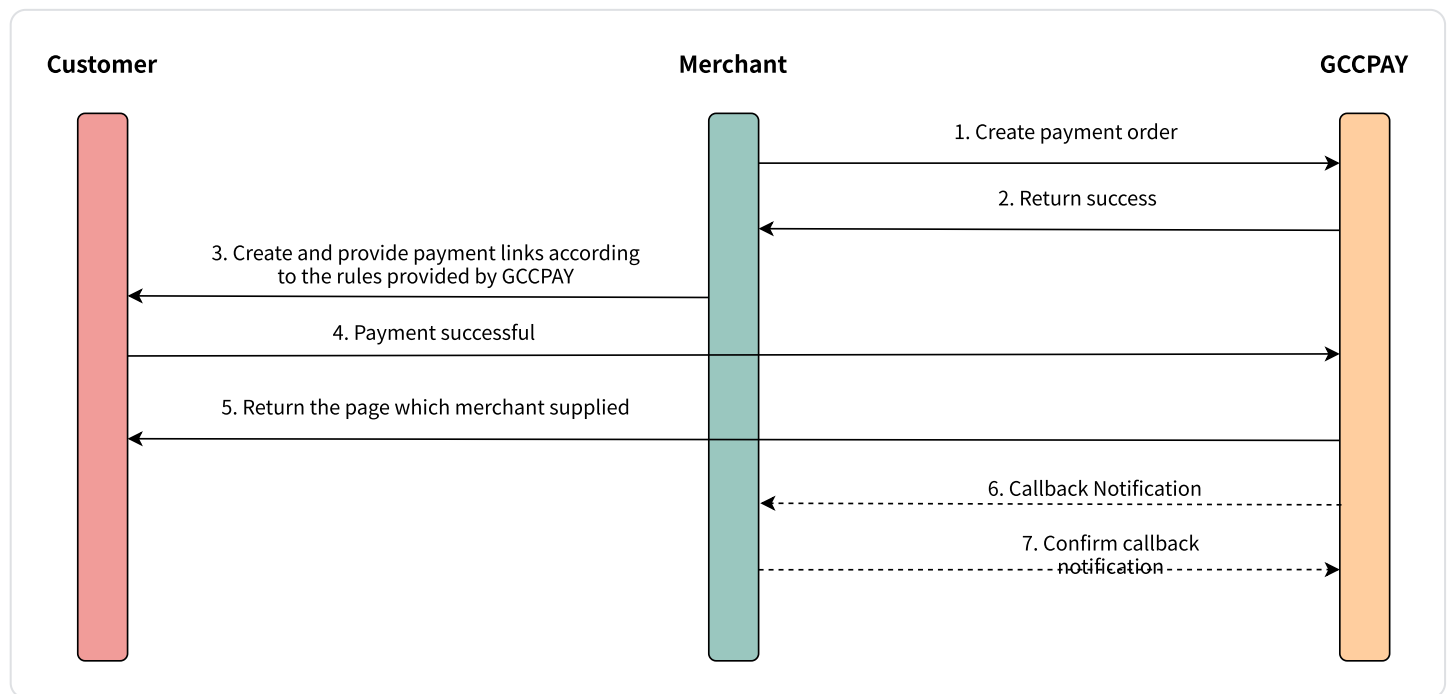
4.1 Create Payment Order

4.1.1 Using GCCPAY Cashier page to create payment order

Summary

- After creating a payment order through the payment system, use the cashier interface provided by GCCPAY to collect money

Sequence diagram



HTTP request type: POST

URL: /merchants/%YOUR_MERCHANT_ID%/orders

- %YOUR_MERCHANT_ID%: Your merchant ID in system
 - For example: if your merchant ID is 'M00001', then the URL is: https://sandbox.gcc-pay.com/api_v1/merchants/M00001/orders

Request

The request parameters for [Create Payment Order] are as follows:

• Header

View public request parameters.

- **Request Body**

Parameters	Type	Required	Describe	Example
merchantOrderId	string	Yes	The merchant payment order ID, can't repeat.	"20221022182334001"
amount	string	Yes	The order amount, floating point number of string, up to 2 digits after the decimal point.	"1.35", "99.10", "127"
currency	string	Yes	Type of currency, and 3 capital letters.	"SAR", "AED", "BHD", "KWD", "OMR", "QAR"
name	string	Yes	The order description.	"Buy a game recharge face value of 100"
notificationURL	string	No	Address for successful order payment notification.	
expiredAt	string	No	The order expires. After the expiration, the order can not be paid. It is recommended to set it to 30 min later, and the default time is 15 days.	"2022-10-23T12:00:21"

- **Request demo**

```
1 {
2   "merchantOrderId": "TEST1672491207",
3   "amount": 12.02,
4   "currency": "SAR",
5   "name": "Order test for desc",
6   "notificationURL": "",
7   "expiredAt": "2023-01-15T20:53:27.000Z"
8 }
```

Response

- The response parameters for [Create Payment Order] are as follows:

Parameters	Type	Description	Sample
id	string	Payment order id	"M448726T20221487632"
clientId	string	Client id	"CLT9208307"
merchantId	string	Merchant id	"M448726"
status	string	Order status <ul style="list-style-type: none"> pending: GCCPAY payment system has created a payment order, but the channel party has not yet created an order doing: GCCPAY payment system and channel order have been created, but the user has not paid paid: User has paid for the order Closed: The order has been closed error: The order has some error locked: The order has been locked expired: The order has expired 	"pending"
ticket	string	Bill voucher, used to verify order ID	"QnTkb04gg5RzEgTW7PYQKCYNPCt8UjsHtBJbBBm1"
name	string	The description of order	"2022-10-23T12:00:00"
merchantOrderId	string	The description of merchant's order	"TEST167249120"
notificationURL	string	Callback address	
amount	float	Payment amount	"12.02"
tip	number	Payment tips	
currency	string	The payment currency	"SAR"
expiredAt	string	The expired time of order	"2023-01-15T20:00:00"

createAt	string	The order created time	"2022-12-31T12:53:17.454Z"
refundAmount	float	The order refund amount	"0"

Response demo

```

1 {
2   "id": "M448726T2022123112531745487632",          ### OrderId
3   "clientId": "CLT9208307",          ### ClientID
4   "merchantId": "M448726",          ### Merchant ID
5   "status": "pending",          ### Order status
6   "ticket": "QnTkb04gg5RzB6fPKLfKH9AJgTW7PYQKCYnP0Wwdz0uiQZjSt8UjsHtBJbBBmTMc"
7   "name": "Order test for desc",          ### Order description
8   "merchantOrderId": "TEST1672491207",          ### Order ID of merchant
9   "notificationURL": "",          ### Callback URL
10  "amount": 12.02,          ### Payment amount
11  "currency": "SAR",          ### Payment currency
12  "tip": 0,          ### Payment tips
13  "expiredAt": "2023-01-15T20:53:27.000Z",          ### Order expired time
14  "createdAt": "2022-12-31T12:53:17.454Z",          ### Order create time
15  "refundAmount": 0          ### Refund amount
16 }
```

The rule of payment link

- Standard payment link URL: `https://sandbox.gcc-pay.com/?orderId={ORDER_ID}&ticket={ORDER_TICKET}&returnURL={RETURN_URL}&language={LANGUAGE}`
- The specific payment link URL:
 - Card Pay: `https://sandbox.gcc-pay.com/mastercard/?orderId={ORDER_ID}&ticket={ORDER_TICKET}&returnURL={RETURN_URL}&language={LANGUAGE}&lockPayMethod=yes`
 - STCPAY: `https://sandbox.gcc-pay.com/stcpay/?orderId={ORDER_ID}&ticket={ORDER_TICKET}&returnURL={RETURN_URL}&language={LANGUAGE}&lockPayMethod=yes`
 - APPLE PAY: `https://sandbox.gcc-pay.com/applePay/?orderId={ORDER_ID}&ticket={ORDER_TICKET}&returnURL={RETURN_URL}&language={LANGUAGE}&lockPayMethod=yes`

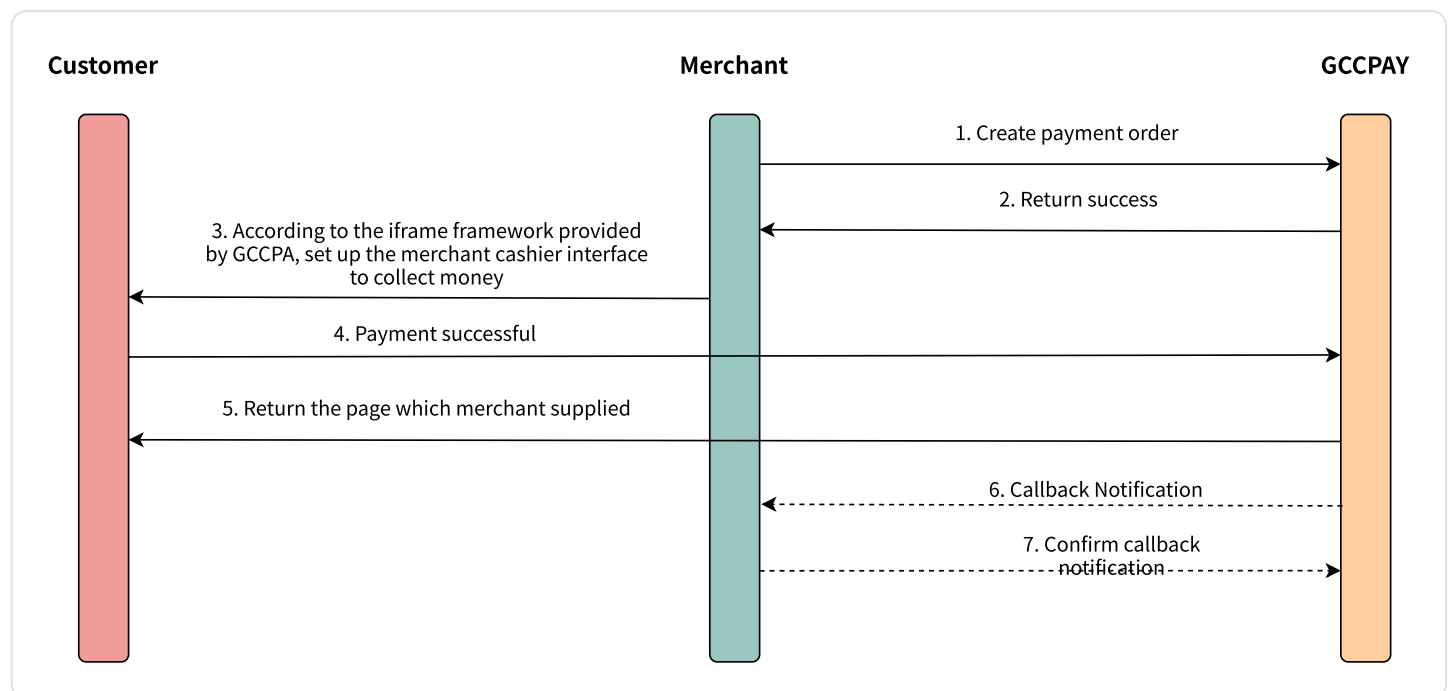
- Specified currency without SAR: `https://sandbox.gcc-pay.com/tap/?orderId={ORDER_ID}&ticket={ORDER_TICKET}&returnURL={RETURN_URL}&language={LANGUAGE}&lockPayMethod=yes`
- Parameter description:
 - orderId: Order ID returned after order created.
 - orderTicket: Ticket returned after order created.
 - returnURL: The page address after payment, and RETURN_URL needs URLencode.
 - language: The default language of the cashier. Optional values are EN(English) and AR(Arabic)
 - lockPayMethod: optional, if the value is 'yes', then it will not show other payment methods on the cashier page

4.1.2 Using iframe to create payment order

Summery

- After creating a payment order through the payment system, use the iframe framework provided by GCCPAY to build your own cashier interface

Sequence diagram



Create payment order

- Create a payment order in GCCPAY payment system by referring to "Create Payment Order"

Create a cash register using the iframe framework

- After successfully creating the payment order, construct the cashier URL according to the returned response parameters

```
https://sandbox.gcc-pay.com/?orderId={ORDER_ID}&ticket={ORDER_TICKET}&returnURL={RETURN_URL}&language={LANGUAGE}
```

- {ORDER_ID}: The order ID returned by GCCPAY after successfully creating the payment order
 - {ORDER_TICKET}: The TICKET returned by GCCPAY after successfully creating the payment order
 - {RETURN_URL}: After the user completes the payment, return to the URL address of the interface, RETURN_ URL needs URLencode
 - {LANGUAGE}: The default language of cashier, optional value is EN(English) and AR(Arabic)
2. Create your own cashier interface. The iframe frame is embedded in the cashier interface. The src of the iframe points to the URL constructed in step 1
- Demo

```
1 var iframe = document.createElement('iframe');
2 iframe.src = 'https://sandbox.gcc-pay.com/?orderId={ORDER_ID}&ticket={ORDER_TICKET}&returnURL={RETURN_URL}&language={LANGUAGE}';
3 iframe.style = 'width: 1000px; height: 600px;';
4 document.body.appendChild(iframe);
```

4.2 Query Payment Order

Summery

- Query the order status in the payment system.

HTTP request type: GET

URL: `/orders/%ORDER_ID%`

- %ORDER_ID%: The order ID which you want to query

Response

【Query payment order】 response as follows:

- **Response Body**

Parameters	Type	Describe	Example
id	string	The payment order ID	"M000001T2022101023455774"
clientId	string	The client ID which created the payment	"CLT0000001"
merchantId	string	The merchant ID to which the payment belong	"M000001"

status	string	Payment order status	"pending"
ticket	string	This is the payment order operation evidence, the URL must contain this parameter when guiding the payer to pay	"Q6XUD1tntwB0QASXzrZesapOs5KqLpySxwYtUfr8TkNNNmI"
name	string	Payment order describe	"Buy a game recharge card with value 100"
channelOrderId	string	Channel ID	
merchantOrderId	string	The merchant payment order ID	"2022102218233400123"
notificationURL	string	Address for successful order payment notification	
amount	number	The order amount. Up to 2 decimal	3.23
currency	string	The currency type. 3 capital letters	"SAR", "AED", "BHD", "EGP", "OMR", "QAR"
expiredAt	string	The order expires. After the expiration, the order can not be paid. It is recommended to set it to 30 min later, and the default time is 15 days.	"2022-10-23T12:00:21.000Z"
refundTimes	number	Times of refunds	0
refundAmount	number	Total refund amount	0.99
message	string	The system message	
createdAt	string	Payment order creates time	"2022-10-22T12:00:21.000Z"
updatedAt	string	Update date	"2022-12-31T12:53:17.000Z"
paidAt	string	Payment date	
payLinkId	string	Pay link ID	"L549641839"
tip	number	The tip in order	"10.02"
payerInfo	string	Payer information	Please refer to the response details
payLink	string	Pay link information	Please refer to the response details

- **Response demo**

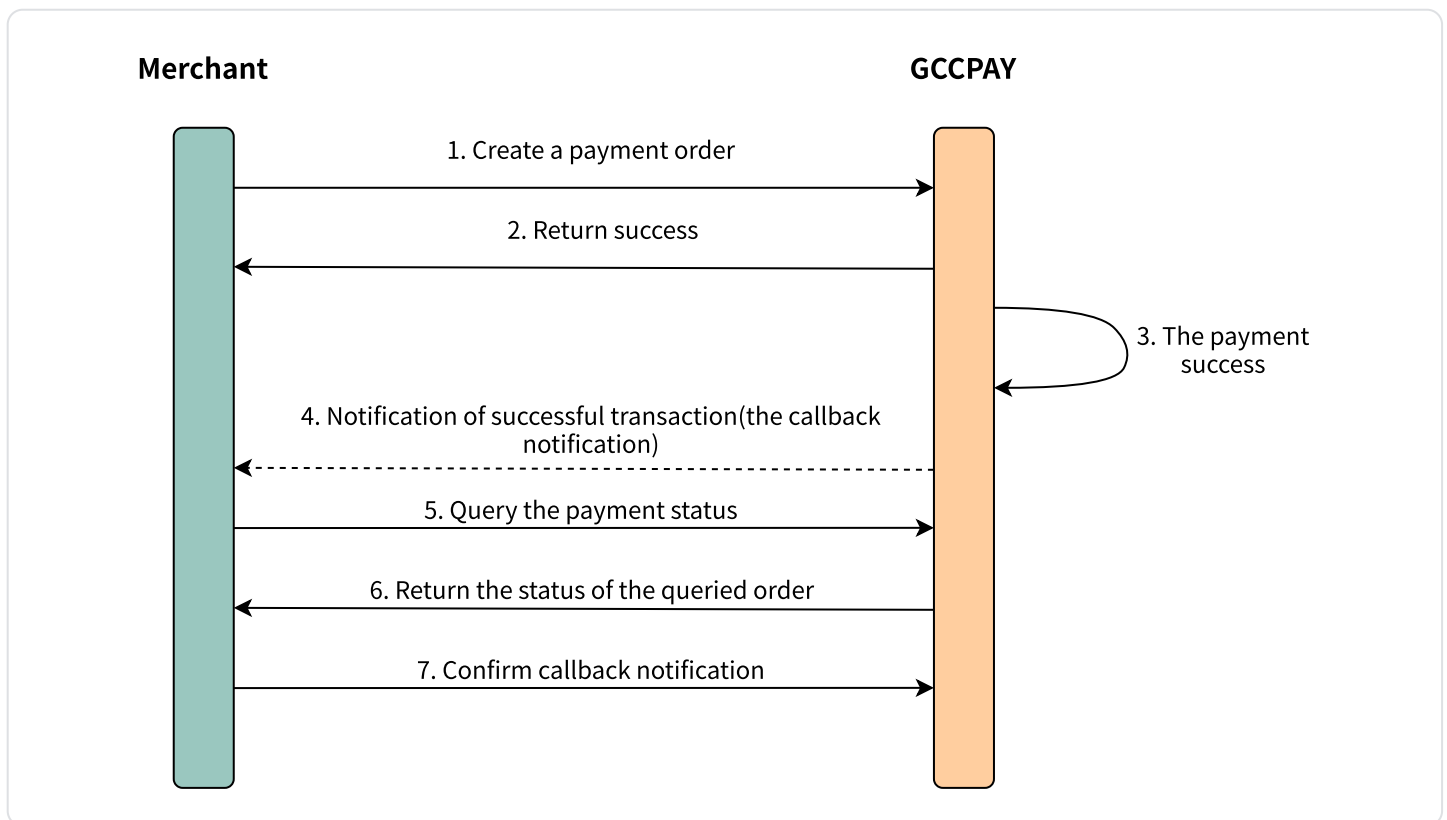
```
1 {
2   "amount": 12.02,          ### Order amount
3   "tip": 0.02,             ### Order tips
4   "refundAmount": 0,        ### Refund amount
5   "id": "M448726T2022123112531745487632",      ### Payment order ID
6   "clientId": "CLT9208307",      ### ClientID
7   "merchantId": "M448726",      ### Merchant ID
8   "payLinkId": "L298791901",     ### payID
9   "status": "pending",          ### Order status
10  "ticket":
    "QnTkb04gg5RzB6fPKlfKH9AJgTW7PYQKCYnP0Wwdz0uiQZjSt8UjsHtBJbBBmTMc",      ###
    Bill voucher, used to verify order ID
11  "name": "Order test for desc",      ### Order description
12  "merchantOrderId": "TEST1672491207",      ### Order ID in merchant
13  "channelOrderId": null,            ### Channel ID
14  "notificationURL": " ",           ### Callback address
15  "currency": "SAR",                ### Payment currency
16  "paidAt": "None",                 ### Payment date
17  "expiredAt": "2023-01-15T20:53:27.000Z",      ### Expired date
18  "message": null,
19  "payerInfo": "
    {\"name\": \"alfa\", \"phone\": \"19884636245\", \"address\": \"alfa\", \"email\": \"a
    lfall@qq.com\"}",      ### Payer information
20  "payLink": "{\"amount\":10,\"tipOptions\":
    [1,2,3],\"id\": \"L296776508\", \"name\": null, \"clientId\": \"CLT7740145\", \"amoun
    tType\": \"fixed\", \"currency\": \"SAR\", \"needContact\": true, \"includeAddress\":
    true, \"needTip\": true, \"tipRequired\": false, \"tipType\": \"percent\", \"creatorId
    \": \"U643029\", \"createdAt\": \"2023-03-01T14:14:00.000Z\", \"updatedAt\": \"2023-
    03-01T14:14:00.000Z\"}",      ### Payment link information
21  "refundTimes": "None",            ### Refund date
22  "createdAt": "2022-12-31T12:53:17.000Z",      ### Create date
23  "updatedAt": "2022-12-31T12:53:17.000Z"      ### Update date
24 }
```

4.3 Callback Notification

Summery

- GCCPAY provides notification callback after order payment completion, and will request the corresponding client

- The merchant needs to provide URL which can receive GET request, GCCPAY will send the callback notification after the payment success. And GCCPAY will attach the `_orderId` parameter to the URL, and the parameter value is the order ID.
- Sequence diagram**



- Merchant uses GCCPAY API to create a payment order
- GCCPAY returned success in creating a payment order
- The payment success
- GCCPAY sends payment success notification to the URL which is provided by merchant
- Merchant uses query order API to confirm the order status
- GCCPAY returns the status of the queried order
- After merchant confirms the payment status, then returns response to callback notification which body is `COMPLETED::%ORDER_ID%`
 - `%ORDER_ID%`: The payment order ID notifications.
 - Demo: `COMPLETED::M448726T2022123112531745487632`

The rules of callback notification

- If merchants do not support the URL callback notification needed, then the callback notification function will not be triggered.
- If GCCPAY does not receive callback notification response, then multiple requests will be made. The rules are as follows:

- The time intervals of the first 10 requests are: 0, 0, 2, 5, 10, 17, 25, 34, 46, 60 second
- After the 11 times, the interval between each subsequent notification is 2500 seconds, and the max number of notifications is 205

4.4 Query Merchant Information

Summery

- Query the merchant information

HTTP request type: GET

URL: `https://sandbox.gcc-pay.com/api_v1/merchants/{%MERCHANT_ID%}`

- {%MERCHANT_ID%}: Merchant ID

Response

【Query Merchant Information】 response as follows:

- **Response Body**

Parameters	Type	Describe	Example
id	string	GCCPAY payment order ID	"M448726"
name	string	Merchant name	"GCCPayMerchant"
status	string	Merchant status: <ul style="list-style-type: none"> • enabled • disabled 	"enabled"
ownerId	string	Login account	"U771297"
name_en	string	User name	
CR_file	string	CR file address	
CR_number	string	CR number	
CR_vatNumber	string	VAT No	
CR_expiredAt	string	CR expired date	
address	string	Merchant adresss	
telephone	string	Merchant telephone	
scope	string	Mercahnt business scope	

legalPerson	string	Merchant legal person	
mcc	string	MCC	"123333"
agencyId	string	Agency ID	"A936252"
currencys	string	Currency which merchants can use	"SAR,KWD,AED,BHDIR"
createdAt	string	Merchant settlement date	"2022-11-08T23:20:"
updatedAt	string	Merchant information update date	"2022-12-07T23:26:"
owner	jsonObject	Merchant owner information including: ID, name and mobile	
addressFile	string	Merchant business address information file URL	
vatFile	string	Merchant vat information file URL	
nationalId	string	Merchant CR file URL	

- **Response demo**

```

1  {
2      "id": "M448726",          ### Merchant ID
3      "name": "GCCPayMerchant",  ### Merchant name
4      "status": "enabled",      ### Merchant status enabled / disabled
5      "ownerId": "U771297",    ### Login account
6      "name_en": "null",       ### Merchant name
7      "CR_file": None,         ### CR file address
8      "CR_number": "null",     ### CR number
9      "CR_vatNumber": "null",   ### VAT Number
10     "CR_expiredAt": None,     ### CR expired date
11     "address": "null",       ### Merchant address
12     "telephone": "null",     ### Merchant telephone
13     "scope": None,          ### Merchant business scope
14     "legalPerson": "null",   ### Merchant legal person
15     "mcc": "123333",        ### MCC
16     "agencyId": "A936252",   ### Agency ID
17     "currencys": "SAR,KWD,AED,BHD,EGP,OMR,QAR",  ### Currency which
    merchants can use
18     "addressFile": null,     ### Merchant business address information file
    URL

```

```

19     "vatFile": null,          ### Merchant vat information file URL
20     "nationalId": null,      ### Merchant CR file URL
21     "createdAt": "2022-11-08T23:20:19.000Z",    ### Merchant settlement
    date
22     "updatedAt": "2022-12-07T23:26:49.000Z",    ### Merchant information
    update date
23     "owner":                ### Merchant owner information
24     {
25         "id": "U771297",      ### login ID
26         "name": "GCCPayMerchant",    ### login name
27         "mobile": "17610908585"    ### login telephone
28     }

```

Appendix

Test Card Information

Card NO	Cardholder Name	Expired Date (MM/YY)	CVV	3D Secure En
512345000000000008	GCCPay_%MerchantName%	07 / 24	100	Y

- %MerchantName% needs to be replaced with the English name of merchant

STCPAY Test Information

Test telephone	Test OTP number
0548220713	12345

Order status

- **pending** : Merchant's payment order is pending
- **paid** : The payer has paid the payment order
- **closed** : The payment order is closed
- **expired** : The payment order is expired

Demo Link

Git : <https://github.com/SGATE-IT/gccpay-client-demo>

Payment Transaction limit

Country	Currency	Minimm payment amount	Maximum payment amount
Saudi Arabia	SAR	0.1	unlimited
Kuwait	KWD	0.25	unlimited
Bahrain	BHD	0.1	3000
The United Arab Emirates	AED	0.1	29000
Oman	OMR	0.1	3000
Qatar	QAR	0.1	29000
Egypt	EGP	0.1	238000

PS: All limits are gateway transaction limits. If the user's bank card has a limit and is less than the gateway transaction limit, the user's bank card limit shall prevail