



**TÀI LIỆU ĐẶC TẢ KỸ THUẬT CÔNG THANH TOÁN  
NỘI ĐỊA GPAY**

Phiên bản 3.0

## Mục lục

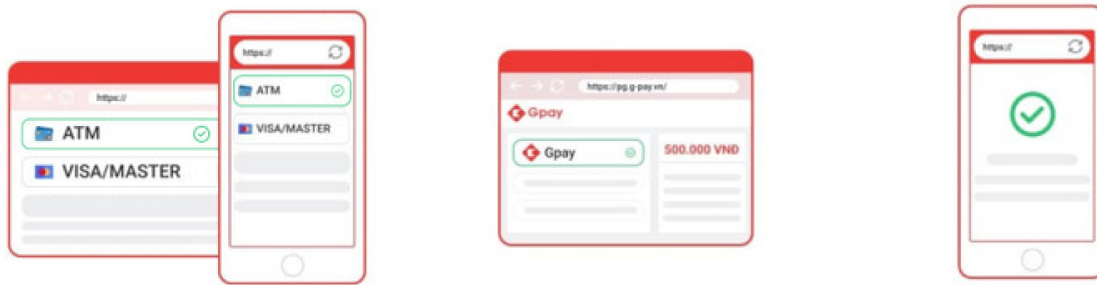
1	Giới thiệu	4
2	Hướng dẫn tích hợp:	5
3	Đặc tả kỹ thuật	6
3.1	Môi trường tích hợp	6
3.2	Phương thức Bảo mật và xác thực	6
3.2.1	Xác thực bằng access Token	6
3.2.2	Xác thực dữ liệu sử dụng chữ ký số	7
3.2.3	SSL	7
3.2.4	API tạo access token	7
3.3	Danh sách các API	8
3.3.1	Tạo đơn hàng (create order)	8
3.3.2	Truy vấn thông tin trạng thái đơn hàng (query order)	10
3.3.3	Xoá Token thẻ	14
3.3.4	Get List Bank	16
3.3.5	Callback/ Webhook nhận kết quả trả về	16
3.4	Response Code	17

## Phiên bản tài liệu

Phiên bản	Ngày	Người tạo	Nội dung thay đổi
3.0	05/2022	Tuan Nguyen Dat	

# 1. Giới thiệu

Tính năng thanh toán qua cổng thanh toán GPAY giúp các Merchant (ĐVCNTT) cho phép khách hàng thanh toán bằng thẻ ATM trên Web hoặc Mobile Web, App của Merchant, sau đó nhập thông tin thẻ để thực hiện thanh toán.



1. End-user chọn thanh toán bằng thẻ ATM

2. End-user thực hiện thanh toán trên Gpay Payment Gateway

3. Thanh toán thành công, merchant web hiển thị kết quả giao dịch

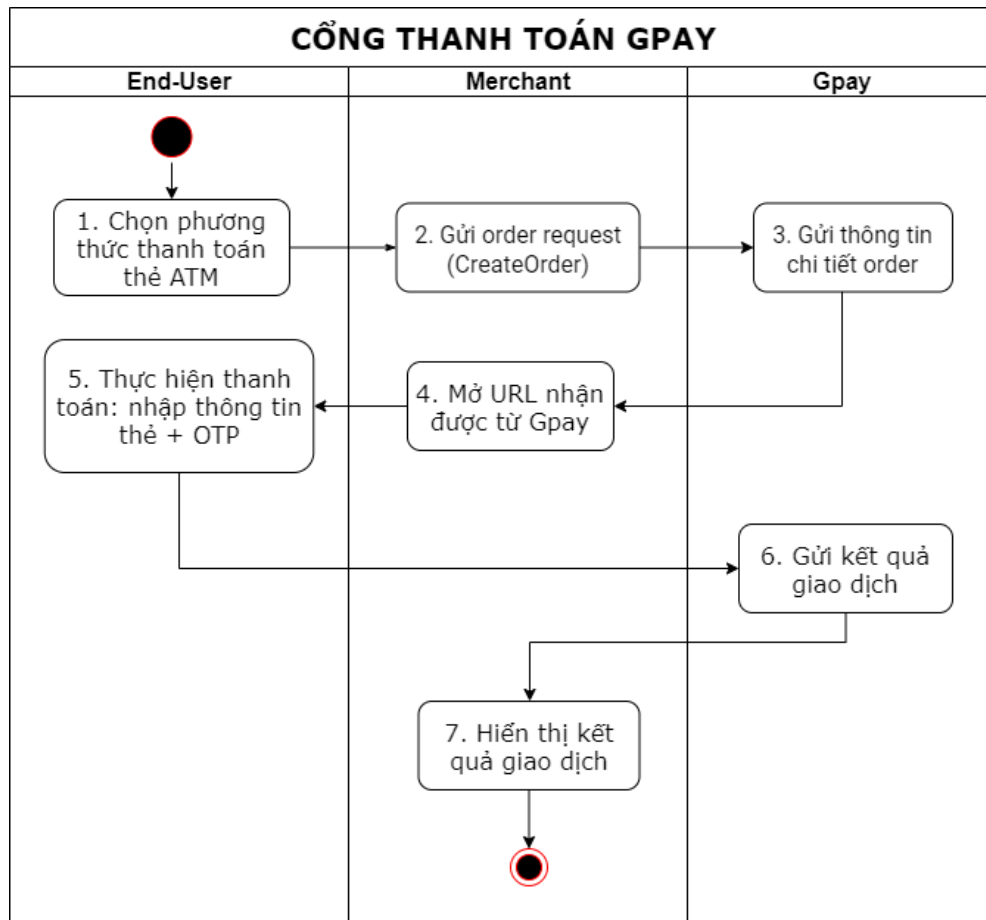


## Giải thích thuật ngữ:

- End-user: là người mua hàng có sử dụng cổng thanh toán do Gpay cung cấp.
- Merchant: là người bán hàng, các cá nhân hoặc đơn vị kinh doanh, cung ứng dịch vụ hoặc/và sản phẩm.
- Gpay: là nền tảng thanh toán điện tử giúp Merchant tích hợp nhiều hình thức thanh toán để bán hàng cho End-user tiện lợi hơn.

## Các bước tích hợp:

- **Bước 1:** End-user chọn hình thức thanh toán bằng ATM trên Web hoặc Mobile Web, App của Merchant.
- **Bước 2:** Merchant gửi yêu cầu tạo đơn thanh toán (createOrder) sang cho GPAY. GPAY trả thông tin đơn hàng về cho Merchant.
- **Bước 3:** Web của Merchant mở link trang thanh toán do GPAY cung cấp để End-user thực hiện bước thanh toán.
- **Bước 4:** End-user thanh toán thành công, GPAY sẽ chuyển về Web của Merchant để hiển thị kết quả giao dịch.



### Các bước thanh toán

- Bước 1: End-user chọn thanh toán bằng thẻ ATM trên website merchant
- Bước 2: Merchant gửi order request (createorder) tới Gpay
- Bước 3: Gpay gửi thông tin order cho merchant
- Bước 4: Merchant mở URL để điều hướng tới Cổng thanh toán Gpay
- Bước 5: Thực hiện thanh toán trên Cổng thanh toán Gpay
- Bước 6: Gpay gửi kết quả thanh toán cho Merchant
- Bước 7: Merchant hiển thị kết quả giao dịch

Tài liệu này được sử dụng bởi các Developer, Technical Personnel và Business Analyst để phục vụ cho việc tích hợp vào Cổng thanh toán của GPAY.

Tài liệu này mô tả toàn bộ các interface, message, data format được sử dụng cho việc tích hợp với cổng thanh toán của GPAY. Đây là tài liệu đặc tả kỹ thuật API và message format.

Toàn bộ các API được mô tả trong tài liệu này bao gồm:

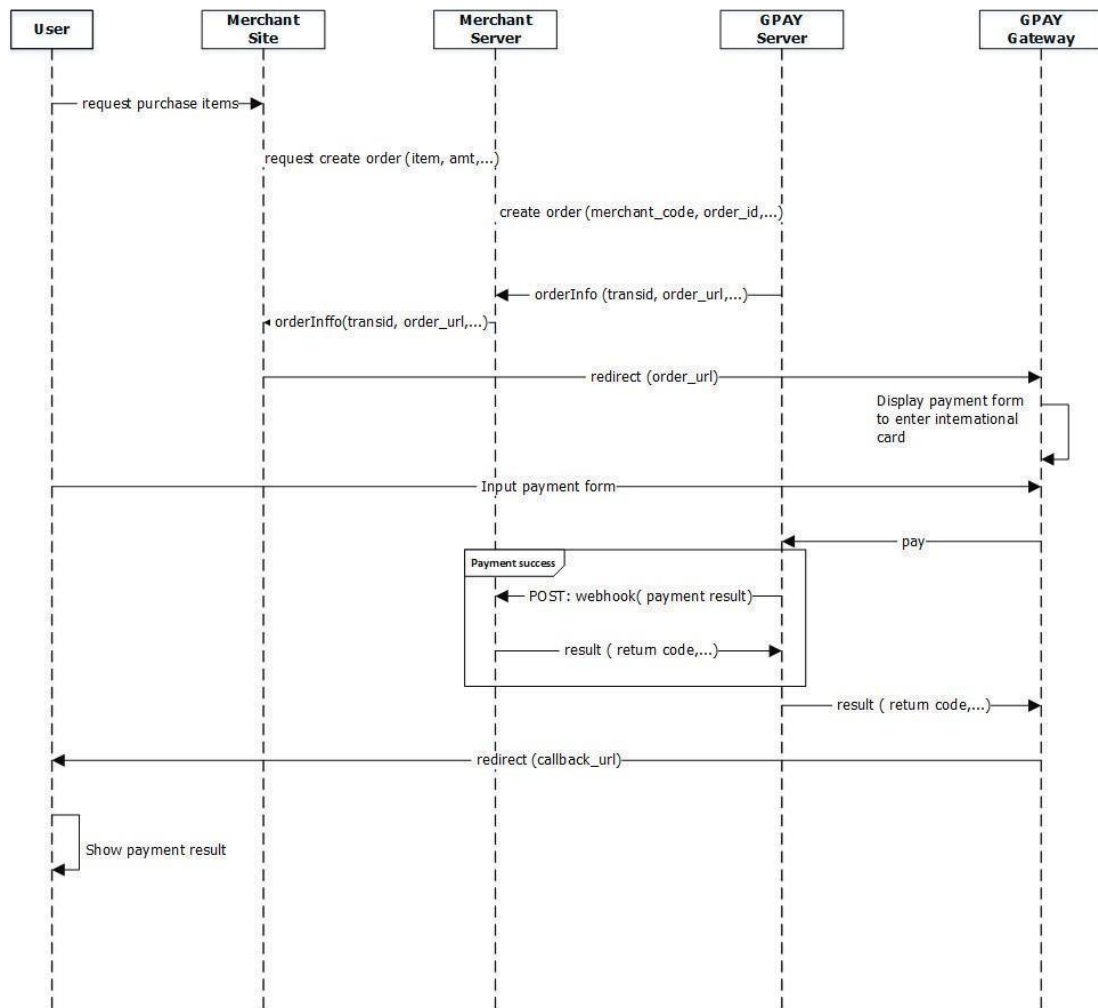
- API tạo access token
- API Tạo đơn hàng (create order)
- API Truy vấn thông tin trạng thái đơn hàng (query order)

Tất cả các Trường dữ liệu trong API được mô tả:

- Tên Trường dữ liệu
- Kiểu trường dữ liệu
- Bắt buộc phải truyền thông tin hay không
- Độ dài: Độ dài tối đa thông tin trường dữ liệu
- Mô tả trường dữ liệu

## 2. Hướng dẫn tích hợp:

### Sơ đồ luồng xử lý:



### **Bước 1: Tạo đơn hàng**

Khi người dùng thực hiện thanh toán trên website của Merchant, website của Merchant sẽ gửi yêu cầu tạo đơn hàng (create order) lên Merchant server. Merchant server sẽ gọi API tạo đơn hàng của GPAY Server

### **Bước 2: Redirect tới trang cổng thanh toán của GPAY**

Khi Merchant server gọi tới API tạo đơn hàng của GPAY server, GPAY server sẽ thực hiện tạo đơn hàng, đồng thời sẽ trả về một đoạn link chuyển tiếp đã được build sẵn gọi là order\_url, Merchant sử dụng order\_url này để redirect người dùng đến trang cổng thanh toán của GPAY.

### **Bước 3. Thực hiện thanh toán**

Người dùng thực hiện nhập các thông tin để thực hiện thanh toán: Số thẻ, Tên chủ thẻ, ngày hết hạn,... sau đó thực hiện các bước thanh toán trên GPAY Gateway, nếu trừ

tiền user thành công GPAY Server sẽ trả về (callback/notify) thông tin trạng thái thanh toán cho Merchant Server theo CallbackURL/Webhook URL đã được gửi từ trước.

#### **Bước 4: Redirect về Merchant Site**

Sau khi thanh toán thành công, GPAY Server sẽ trả về trạng thái giao dịch cho GPAY Gateway. GPAY Gateway chuyển hướng (redirect) về Merchant Site để hiển thị kết quả giao dịch trên giao diện.

## **2 Đặc tả kỹ thuật**

### **3.1 Môi trường tích hợp**

#### **1) Thông tin xác thực merchant:**

- merchant\_code: Sẽ được cấp riêng cho merchant
- password : Sẽ được cấp riêng cho merchant

#### **2) Thông tin key**

- public key xác thực chữ ký số

#### **3) Thông tin thẻ test**

Thẻ nội địa test:

- Số thẻ: 9704 0000 0000 0018
- Tên chủ thẻ: Nguyen Van A
- Ngày hiệu lực: 03/07
- Xác thực OTP: OTP

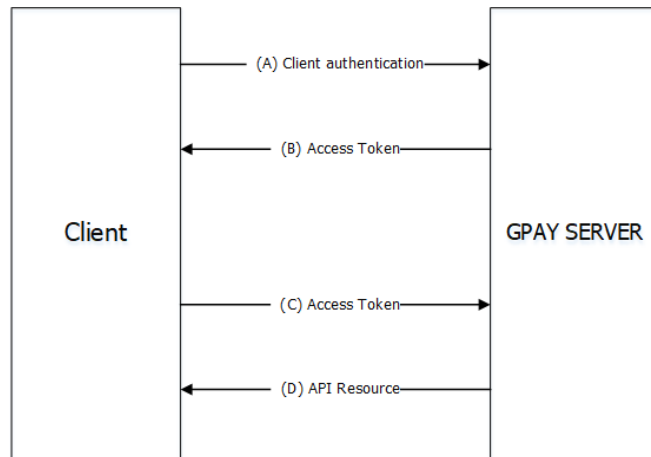
### **3.2 Phương thức Bảo mật và xác thực**

#### **3.2.1 Xác thực bằng access Token**

Các API của GPAY sử dụng Phương thức xác thực bằng access Token.

#### **Luồng xử lý:**





A – Merchant gửi request để tạo access\_token tới GPAY Server.

B – GPAY Server sẽ xác thực Merchant, nếu xác thực thành công sẽ tạo và trả về cho Merchant access\_token.

C – Merchant sử dụng access\_token để gọi các API của GPAY Server

D – GPAY Server sẽ kiểm tra access\_token và phục vụ các lời gọi của Merchant.

### 3.2.2 Xác thực dữ liệu sử dụng chữ ký số

Để đảm bảo dữ liệu hai bên không bị giả mạo, hai bên sẽ sử dụng chữ ký số trên dữ liệu trao đổi. Quy tắc sinh chữ ký số (trường signature) được thực hiện theo công thức sau:

```
openssl_sign($input, $binary_signature, $privateKey, OPENSSSL_ALGO_SHA256);
```

```
$signature = base64_encode($binary_signature);
```

Dữ liệu trường input theo cấu trúc:

```
INPUT: Field_Name1=Field_Value1& Field_Name2=Field_Value2.....
```

Danh sách các Field để sinh chữ ký số sẽ được đặc tả theo từng API

*Vi dụ INPUT: merchant\_code={}&order\_id={}&amount={}*.

- *merchant\_code*: Mã merchant
- *order\_id*: Mã đơn hàng
- *amount*: tổng số tiền đơn hàng

*INPUT: merchant\_code=MERCHANT\_XYZ&transaction\_id=12345678&amount=10000*

### 3.2.3 SSL

Kết nối giữa Merchant và GPAY sử dụng giao thức kết nối HTTPS với giao thức TLS1.2; toàn bộ dữ liệu sẽ được bảo mật mã hóa trong quá trình truyền dữ liệu.

### 3.2.4 API tạo access token

URL	<a href="/authentication/token">/authentication/token</a>
HTTP Method	POST
Authentication	<p>Phương thức bảo mật và xác thực được sử dụng:</p> <ul style="list-style-type: none"> <li>• Chữ ký số.</li> <li>• SSL</li> </ul>
Protocol	HTTPS/REST-JSON

ST T	Trường dữ liệu	Kiểu dữ liệu	Bắt buộc	Độ dài	Mô tả
<b>API Input</b>					
1	merchant code	String	Y	20	Mã Merchant
2	password	String	Y	20	password được cấp cho merchant
<b>API Output</b>					
1	meta.code	string	Y	10	Mã trạng thái của API
2	meta.msg	string	Y	256	Mã lỗi
3	meta.internal_msg	string	Y	256	Mã lỗi chi tiết
4	response.token	string	Y	256	Token được tạo để merchant sử dụng cho lần gọi các API tiếp theo. Trong trường hợp Token hết hạn cần gọi lại hàm tạo token để lấy token mới
5	response.expired_at	timestamp	Y		Thời gian hết hạn của token(UTC TIME)

### 3.3 Danh sách các API

#### 3.3.1 Tạo đơn hàng (Init order)

URL	/order/init
HTTP Method	POST
Header	Authorization: Bearer {token lấy từ response của <b>API tạo access token</b> ( <a href="/authentication/token">/authentication/token</a> )}
Authentication	<p>Phương thức bảo mật và xác thực được sử dụng:</p> <ul style="list-style-type: none"> <li>• Sử dụng Access Token</li> <li>• Chữ ký số.</li> <li>• SSL</li> </ul>

Protocol	HTTPS/REST-JSON
----------	-----------------

STT	Trường dữ liệu	Kiểu dữ liệu	Bắt buộc	Độ dài	Mô tả
<b>API Input</b>					

1	merchant_code	String	Y	20	Mã Merchant
2	order_id	String	Y	50	Mã giao dịch của đơn hàng. Mã giao dịch phải là duy nhất. Mã giao dịch có thể theo format: YYYYMMDD-ORDER_ID; trong đó ORDER_ID là số tự tăng
3	order_amt	int	Y		Tổng giá trị đơn hàng theo VND
4	order_currency	String	Y	3	Đơn vị tiền của đơn hàng, mặc định là VND
5	order_title	String	N	256	Tiêu đề đơn hàng
6	order_description	String	N	256	Mô tả đơn hàng
7	order_time	int	Y		Thời gian tạo đơn hàng (tính đến milisecond)
8	order_item	String	N	2048	Danh sách hàng hóa của đơn hàng do merchant tự định nghĩa, có thể theo format như sau: [{"itemid":"","itemname":"","itemprice":"","itequantity":""}]
9	bank_code	String	N	20	Mã ngân hàng(sử dụng gpay_bank_code trong api /order/bank-list)
10	customer_name	String	N	256	Tên khách hàng
11	customer_id	String	N	256	Mã khách hàng
12	phone	String	N	20	Số điện thoại của khách hàng
13	email	String	N	20	Email khách hàng
14	address	String	N	1024	Địa chỉ khách hàng
15	callback_url	String	Y	1024	URL của merchant được redirect khi kết quả thanh toán thành công hoặc thất bại.
16	webhook_url	String	Y	1024	webhook_url của merchant được gọi khi kết quả thanh toán thành công hoặc thất bại.
17	use_custom_domain	Boolean	N		
18	language	String	Y	2	Ngôn ngữ hiển thị của trang thanh toán Gpay, mặc định vi
19	service_code	String	Y	50	Mặc định: <b>PAYMENTGATEWAY</b>
20	expired_order	Number	N		Thời gian order expired tính theo millisecond <i>default: 600000</i> <i>minimum: 30000</i> <i>maximum: 900000</i>
21	voucher_info	String	N	256	Thông tin mã giảm giá (hiện tại chưa sử dụng)

22	payment_method	String	Y	50	Phương thức thanh toán: <i>default: <b>BANK_ATM</b></i> [ BANK_ATM, PAY_BY_VOUCHER, BUY_NOW_PAY_LATER, PAY_BY_INSTALLMENT ]
23	payment_type	String	Y	50	Loại thanh toán: <i>default: <b>IMMEDIATE</b></i> [ IMMEDIATE, DELAYED ]
24	embed_data	String	Y	2048	Dữ liệu riêng của transaction do merchant truyền vào
25	token_id	String	N	256	token for card of customer(by customer id). Trường hợp truyền token thẻ thì sẽ không cần nhập thông tin thẻ.
26	signature	String	Y	2048	Chữ ký số checksum được tạo theo công thức sinh chữ ký được mô tả trong mục Bảo mật và phương thức xác thực. Giá trị trường <b>INPUT</b> như sau: <b>INPUT:merchant_code={}&amp;order_id={}&amp;order_amt={}&amp;embed_data={}&amp;order_currency={}&amp;language={}</b> . • merchant_code: Mã merchant • order_id: Mã đơn hàng • order_amt: giá trị đơn hàng. • embed_data: dữ liệu riêng của merchant • language: ngôn ngữ • order_currency: loại tiền tệ. Ví dụ: merchant_code=MERCHANT_XYZ&order_id=order1234&order_amt=10000&embed_data=private_data&order_currency=VND&language=vi
<b>API Output</b>					
1	meta.code	string	Y	10	Trạng thái xử lý của API
2	meta.msg	string	Y	256	Mã thông báo
3	meta.internal_msg	string	Y	256	Mã thông báo chi tiết
4	response.order_url	string	Y		Sử dụng giá trị url này để redirect tới trang thanh toán của Gpay payment gateway.
5	response.gpay_trans_id	string	Y	50	Gpay transaction id. Đây là trường mã giao dịch được Gpay tạo ra và là duy nhất ứng với mỗi đơn hàng. Mã giao dịch này sẽ được sử dụng trong các API truy vấn trạng thái xử lý đơn hàng và thông tin của đơn hàng
6	response.signature	string	Y	2048	Chữ ký số checksum được tạo theo công thức sinh chữ ký được mô tả trong mục Bảo mật và phương thức xác thực. Giá trị trường INPUT như sau: gpay_trans_id={}&order_id={}

### 3.3.2 Truy vấn thông tin trạng thái đơn hàng (query order)

URL	/order/detail
HTTP Method	POST
Header	Authorization: Bearer {token lấy từ response của API <b>API tạo access token</b> }
Authentication	<p>Phương thức bảo mật và xác thực được sử dụng:</p> <ul style="list-style-type: none"> <li>• Sử dụng Access Token</li> <li>• Chữ ký số.</li> <li>• SSL</li> </ul>
Protocol	HTTPS/REST-JSON

STT	Trường dữ liệu	Kiểu dữ liệu	Bắt buộc	Độ dài	Mô tả
<b>API Input</b>					
1	merchant_code	String	Y	20	Mã Merchant
2	gpay_trans_id	string	Y	50	Gpay transaction id. Đây là trường mã giao dịch được Gpay tạo ra và là duy nhất ứng với mỗi đơn hàng. Mã giao dịch này sẽ được sử dụng trong các API truy vấn trạng thái xử lý đơn hàng và thông tin của đơn hàng
3	signature	String	Y	2048	Chữ ký số checksum được tạo theo công thức sinh chữ ký được mô tả trong mục Bảo mật và phương thức xác thực. Giá trị trường <b>INPUT</b> như sau: <b>INPUT:</b> merchant_code={}&gpay_trans_id={}
<b>API Output</b>					
1	meta.code	string	Y	10	Trạng thái xử lý của API
2	meta.msg	string	Y	256	Mã thông báo
3	meta.internal_msg	string	Y	256	Mã thông báo chi tiết
4	response.order_url	string	Y		Sử dụng giá trị url này để redirect tới trang thanh toán của Gpay payment gateway.
5	response.gpay_trans_id	string	Y	50	Gpay transaction id. Đây là trường mã giao dịch được Gpay tạo ra và là duy nhất ứng với mỗi đơn hàng. Mã giao dịch này sẽ được sử dụng trong các API truy vấn trạng thái xử lý đơn hàng và thông tin của đơn hàng
6	response.order_id	String	Y	50	Mã giao dịch của đơn hàng. Mã giao dịch phải là duy nhất. Mã giao dịch có thể theo format: YYYYMMDD-ORDER_ID; trong đó ORDER_ID là số tự tăng
7	response.order_amt	int	Y		Tổng giá trị đơn hàng theo VND

8	response.order_status	String	Y	20	<p>Trạng thái xử lý của đơn hàng</p> <ul style="list-style-type: none"> <li>- <b>ORDER_PENDING</b>: Đơn đặt hàng được bắt đầu và đang chờ xử lý</li> <li>- <b>ORDER_PROCESSING</b>: Đơn hàng thanh toán đang trong quá trình xử lý</li> <li>- <b>ORDER_SUCCESS</b>: Đơn hàng thanh toán thành công</li> <li>- <b>ORDER_FAILED</b>: Đơn hàng thất bại</li> <li>- <b>ORDER_CANCEL</b>: Đơn hàng bị hủy</li> <li>- <b>ORDER_VERIFYING</b>: Đơn hàng đang được xác thực</li> </ul>
9	response.signature	string	Y	2048	<p>Chữ ký số checksum được tạo theo công thức sinh chữ ký được mô tả trong mục Bảo mật và phương thức xác thực. Giá trị trường INPUT như sau:  INPUT: gpay_trans_id={}&amp;order_id={}&amp;order_amt={}&amp;response.order_status={}</p> <ul style="list-style-type: none"> <li>• merchant_code: Mã merchant</li> <li>• order_id: Mã đơn hàng</li> <li>• order_amt: Giá trị đơn hàng</li> <li>• order_status: Trạng thái đơn hàng</li> </ul> <p>Ví dụ:  merchant_code=MERCHANT_XYZ&amp;transaction_id=1234 5678&amp;order amt=10000&amp;order status=0</p>

### 3.3.3 Token thẻ (remove card token)

URL	/order/remove-token
HTTP Method	POST
Header	Authorization: Bearer {token lấy từ response của API <b>API tạo access token</b> }
Authentication	<p>Phương thức bảo mật và xác thực được sử dụng:</p> <ul style="list-style-type: none"> <li>• Sử dụng Access Token</li> <li>• Chữ ký số.</li> <li>• SSL</li> </ul>
Protocol	HTTPS/REST-JSON

STT	Trường dữ liệu	Kiểu dữ liệu	Bắt buộc	Độ dài	Mô tả
<b>API Input</b>					
1	merchant_code	String	Y	20	Mã Merchant
2	token_id	string	Y	256	Token thẻ được tạo khi init Order
3	signature	String	Y	2048	Chữ ký số checksum được tạo theo công thức sinh chữ ký được mô tả trong mục Bảo mật và phương thức xác thực. Giá trị trường <b>INPUT</b> như sau: <b>INPUT:</b> merchant_code={}& token_id={}
<b>API Output</b>					
1	meta.code	string	Y	10	Trạng thái xử lý của API
2	meta.msg	string	Y	256	Mã thông báo
3	meta.internal_msg	string	Y	256	Mã thông báo chi tiết
4	response.merchant_code	String	Y	20	Mã Merchant
5	response.token_id	string	Y	256	Token thẻ được tạo khi init Order
6	response.signature	string	Y	2048	Chữ ký số checksum được tạo theo công thức sinh chữ ký được mô tả trong mục Bảo mật và phương thức xác thực. Giá trị trường <b>INPUT</b> như sau: <b>INPUT:</b> merchant_code={}& token_id={}

### 3.3.4 Get List Bank

URL	/order/bank-list
HTTP Method	GET
Authentication	None
Protocol	HTTPS/REST-JSON

STT	Trường dữ liệu	Kiểu dữ liệu	Bắt buộc	Độ dài	Mô tả
<b>API Input</b>					
<b>API Output</b>					
1	meta.code	string	Y	10	Trạng thái xử lý của API
2	meta.msg	string	Y	256	Mã thông báo
3	meta.internal_msg	string	Y	256	Mã thông báo chi tiết
4	response.bank_name	String	N	20	Tên ngân hàng
5	gpay_bank_code	string	N	256	Mã ngân hàng của GPAY (có thể dùng trường này cho params bank_code trong API tạo order: <b>/order/init</b> )
6	short_name_by_napas	string	N	256	Mã ngân hàng theo Napas
7	short_name_by_gpay	string	N	256	Tên ngân hàng theo GPAY
8	pub_lich_name	string	N	256	Tên ngân hàng
9	bank_bin	string	N	256	Mã PIN ngân hàng
10	provider	string	N	256	Ngân hàng
11	status	string	N	256	Trạng thái
12	logo	string	N	256	Ảnh logo ngân hàng

### 3.3.5 Callback/ Webhook nhận kết quả trả về từ GPAY

**API:** Merchant cần dựng api này để xử lý kết quả trả về từ Gpay. Merchant sẽ trả về http status 200 để thông báo cho Gpay kết quả giao dịch thành công

STT	Trường dữ liệu	Kiểu dữ liệu	Bắt buộc	Độ dài	Mô tả
<b>API Input</b>					
1	order_id	String	Y	20	Order Id của merchant
2	gpay_trans_id	string	Y	50	Gpay transaction id. Đây là trường mã giao dịch được Gpay tạo ra và là duy nhất ứng với mỗi đơn hàng. Mã giao dịch này sẽ được sử dụng trong các API truy vấn trạng thái xử lý đơn hàng và thông tin của đơn hàng
3	status	String	Y	20	Trạng thái xử lý của đơn hàng - <b>ORDER_PENDING:</b> Đơn đặt hàng được bắt đầu và đang chờ xử lý - <b>ORDER_PROCESSING:</b> Đơn hàng thanh toán đang trong quá trình xử lý - <b>ORDER_SUCCESS:</b> Đơn hàng thanh toán thành công - <b>ORDER_FAILED:</b> Đơn hàng thất bại - <b>ORDER_CANCEL:</b> Đơn hàng bị hủy



					- <b>ORDER VERIFYING</b> : Đơn hàng đang được xác thực
4	embed_data	String	Y	2048	Dữ liệu riêng của giao dịch
5	token_info	String	N	2048	Thông tin token của thẻ được mã hoá base64 Ví dụ: object: { "id": 1234 "bank_code": "GPVCCB", "card_number": "9704...." }
6	signature	String	Y	2048	Chữ ký số checksum được tạo theo công thức sinh chữ ký được mô tả trong mục Bảo mật và phương thức xác thực. Giá trị trường INPUT như sau: <b>INPUT</b> : <b>order_id={}&amp;gpay_trans_id={}&amp;status={}</b>
<b>API Output: response http code status is 200 if success</b>					

### 3.4 Response Code

Thông tin trạng xử lý của API được trả về qua trường **meta.code**.

Danh sách các mã trả về của trường này như sau:

**meta.code = 200 => API xử lý thành công meta.code #**

**200 => API xử lý lỗi**

**Mã lỗi:** Danh sách mã lỗi được gửi đính kèm cùng tài liệu