



Shopee Open API 2.0 Introduction

Open API 2.0接口介紹

2021.05 Latest update by Ethel



基本說明	3
背景介紹	3
v2建立基礎	3
v2功能說明	3
v2重點功能	3
v2 APP類型	4
v2接口測試	7
v2測試環境	7
v2賣場授權	7
簽名計算與授權連結組成步驟	8
授權逾時邏輯	9
取得code與shop_id/ main_account_id	10
取消賣場授權	10
v2接口驗證與請求	11
API通訊協定	11
請求/呼叫方式	11
使用參數	11
Access Token	15
GetAccesstoken	15
RefreshAccessToken	18
計算簽名	21
基本字串	21
計算簽名	21



基本說明

背景介紹

目前Open API 1.0 (v1)已經開放一段時間，並提供給眾多開發者使用，然而部分v1接口存在技術上的限制，導致開發者在開發時可能會遇到一些潛在的問題。為此開放平台將優化接口性能，並且增加更多應用場景，以下簡述v1問題：

- 沒有針對不同功能模組進行授權機制的拆分。
- 接口標準化程度不足，包含：接口名稱、欄位名稱、描述、錯誤訊息……等。
- 現有接口使用場景不足以涵蓋多數業務需求，開發者必須搭配賣家中心合併使用。

v2建立基礎

- v2會涵蓋所有v1的功能。
- 開發者(同一個開發者帳號下的所有APPs與Partner IDs)可以在轉換期針對同一個功能模組，同時使用v1與v2的接口。
- 開發者必須在轉換期結束以前，以模組維度(**module**)決定使用v1或是v2的接口。轉換期結束後，不存在接口並存的使用場景。
選擇v1的開發者僅能使用v1接口，但後續可以將接口升級成v2；
選擇v2的開發者僅能使用v2接口，且後續無法將接口降級成v1。
- 具體轉換期起訖時間將於後續平台公告。

v2功能說明

v2重點功能

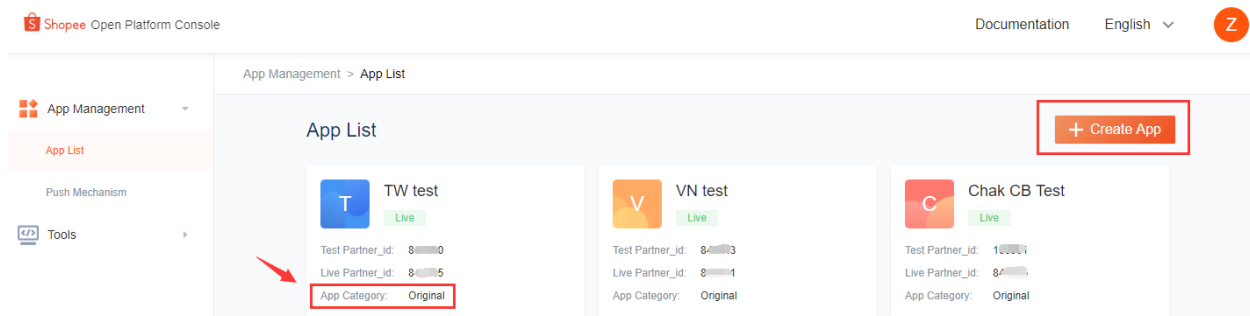
- 新的授權與驗證方式
 - a. 為了保護開放平台遭受可能的風險，並確保開發者的帳戶安全，v2將使用新的授權方式與驗證方式。具體內容可以參考後續說明。
 - b. 除了賣場維度的授權，v2同時也支持主帳號(main account)維度的授權。藉此，merchant賣家可以一次將主帳號賣場授權後，其下歸屬的子賣場也同時授權。具體內容可以參考[Developer Guide](#)。
- 提供兩種請求方式：GET與POST，和兩種參數形式：公共參數(common parameters)與請求參數(request parameters)。具體內容可以參考後續說明。
- 根據不同的開發者帳戶類型區分不同的APP可以使用的API接口，具體內容可以參考[Developer Types and App Types](#)或後續說明。
- Media module支持影片與圖片上傳，請求參數支持file type的格式。具體內容可以參考相關的接口文件。
- Chat module管理買賣家之間的溝通 (目前TW用戶不支援)



- Listing module
 - 拆分建立商品與建立規格的步驟。商品必須先被建立後，再使用tier-variation相關的接口建立規格。具體內容可以參考相關的接口文件。
 - 屬性(Attribute)支持兩種新的輸入格式，可以同時輸入多個值：
 - MULTIPLE_SELECT
 - MULTIPLE_SELECT_COMBO_BOX
 - 建立商品或更新商品時提供新的屬性欄位: value_id
- Order/Fulfillment module
 - 整合訂單維度和包裹維度(order split)的資訊在同一接口
 - 優化get_order_details接口的指令回覆參數(response parameter)，縮短響應時間以適用於不同場景
 - 提供新的訂單狀態:PROCESSED，此狀態代表訂單已安排出貨。過去"已安排出貨但尚未被物流業者掃寄件條碼"與"未安排出貨"的狀態皆歸屬在READY_TO_SHIP。具體內容可以參考[API Call Flows Order Status Flow](#)。
 - 安排出貨流程與列印寄件單流程更新為異步流程。開發者必須請求多個API接口完成安排出貨並取得寄件編號後，再請求列印寄件單的相關API接口
 - 寄件單取得方式變更為檔案格式(file type)，而非URL
- Payment module
 - 整合v1的GetEscrowDetails接口與 MyIncome接口
 - 針對促銷組合Bundle Deal與加價購Add-on Deal訂單，escrow 接口會拆分折扣價格至商品維度(item level)
 - 針對中國跨境 SIP 訂單進帳資訊支持提供主賣場與子賣場更多貨幣欄位
- 優化錯誤訊息讓開發者更明確的了解請求錯誤原因

v2 APP類型

請求v2接口以前，開發者必須檢查APP的類型，如下圖所示：



若APP Category顯示"Original"，則此類型的APP不支援v2的接口，請建立新的APP。建立新的APP後則可取得新的partner_id，開發者必須先執行賣場授權，待賣場授權到新的APP後，方可請求v2接口。

不同的開發者類型可以建立的APP類型會有不同，具體差異請參考[Developer Types and App Types](#)或下表



開發者類型	可建立APP類型	該類APP可使用的OpenAPI
既有開發者 (Original Developer)	Original	No API
第三方企業開發者 (Enterprise 3rd Party)	ERP System	All v2 APIs 除了Customer Service
	Product Management	All v2 Shop APIs All v2 ShopCategory APIs All v2 Item APIs All v2 Public APIs All v2 Media Space APIs Push APIs: <ul style="list-style-type: none"> Promotion Update Push Item Promotion Info Push Reserved Stock Change Push Banned Item Push Shopee Updates Puch
	Order Management	All v2 Order APIs All v2 Logistics APIs All v2 FirstMileTracking APIs All v2 Public APIs Other APIs: <ul style="list-style-type: none"> v2.product.get_item_base_info v2.product.get_item_promotion v2.product.get_comment v2.product.get_discount Push APIs: <ul style="list-style-type: none"> Order Status Push Order TrackingNo Push Shopee Updates Push
Accounting and Finance	All v2 Payment APIs All v2 Public APIs Other APIs: <ul style="list-style-type: none"> v2.order.get_order_detail v2.payment.get_escrow_detail v2.payment.get_escrow_list v2.product.get_item_base_info v2.product.get_item_promotion v2.product.get_comment v2.product.get_discount Push API: <ul style="list-style-type: none"> Shopee Updates Push 	



	Marketing	<p>All v1 TopPicks APIs All v1 Discount APIs All v1 Public APIs Other APIs:</p> <ul style="list-style-type: none"> • v2.product.get_item_base_info • v2.product.get_item_promotion <p>Push APIs:</p> <ul style="list-style-type: none"> • Promotion Update Push • Item Promotion Info Push • Reserved Stock Change Push • Banned Item Push • Shopee Updates Push
	Customer Service	<p>All Chat APIs Other APIs:</p> <ul style="list-style-type: none"> • v2.shop.get_shop_info • v2.product.get_item_list • v2.product.get_item_base_info • v2.order.get_order_list • v2.order.get_order_detail • v2.payment.get_escrow_detail • v2.logistics.get_tracking_info <p>Push APIs</p> <ul style="list-style-type: none"> • Shopee Updates Push
商城賣家 (Enterprise Seller)	Seller In-house System	All v2 APIs 除了Customer Service

請注意：

1. TW僅開放給”第三方企業開發者”和”商城賣家”
不包含”第三方個人開發者 Individual 3rd Party”與”個人賣家 Individual Seller”
2. Customer Service類型目前不支援



v2接口測試

v2測試環境

開發者進行v2接口串接以前，請務必使用新的沙盒(sandbox)先進行測試後，再進行後續的正式環境串接。

請於Open Platform>Console>Tools>Test Shop>Create Test Shop建立測試賣場，每一組測試賣場會提供一組對應的買家帳號，以便您創建測試訂單。如果您需要測試相關的資訊，如：商品、訂單，請自行建立。更多內容可以參考[Developer Guide](#)

The screenshot shows the 'Shopee Open Platform Console' interface. The top navigation bar includes 'Documentation' and 'English' with a dropdown arrow, and a user profile icon 'Z'. The left sidebar contains 'App Management' and 'Tools', with 'Test Shop' highlighted in a red box. The main content area is titled 'Tools > Test Shop' and features a '+ Create Test Shop' button in a red box. Below this, a card displays the following information:

Shop ID: 53	More :
Shop Account	SANDBOX.b98: [redacted] 49b0d
Shop Password	b60 [redacted] 85cb
URL	https://test-stable.shopee.com.br/shop/53
Area	Local - BR
Expiration Date	20-03-2021 00:00
Buyer Account	
Buyer Account	SANDBOX_BUYER.6 [redacted] 3383
Buyer Password	036 [redacted] 484

v2賣場授權

欲執行v2賣場授權，開發者必須先建立授權連結後，並將授權連結提供給賣家，賣家輸入賣場帳密後方可完成授權。請注意，授權連結請置入於開發者的系統中，以方便進行賣家操作。

開發者可以透過測試環境賣場帳密，用以模擬賣家授權流程。



授權連結是一串將授權URL、必要參數等資訊帶入的URL。

授權URL

所屬環境	HTTP Address
URL	https://partner.shopeemobile.com/api/v2/shop/auth_partner
Test URL	https://partner.test-stable.shopeemobile.com/api/v2/shop/auth_partner

必要參數

欄位	類型	必要	說明
partner_id	int	yes	partner_id可以透過進入Console後，選擇APP，在APP中則可取得該資訊
redirect	string	yes	轉跳URL。此連結用於授權成功後的轉跳頁面，可以使用任何網址，例如公司官網、賣場網址
sign	string	yes	根據Shopee開放平台規則加密計算出的簽名字串
timestamp	int	yes	時間戳必須與計算簽名(sign)的時間一致，且會於5分鐘後過期失效

指令回覆參數

欄位	類型	必要	說明
code	int	yes	用此獲取access_token和refresh_token。僅能一次性使用，且於10分鐘後過期失效
shop_id	string	no	授權給開發者的賣場shop_id
main_account_id	string	no	授權給開發者的主帳號main_account_id

簽名計算與授權連結組成步驟

此部分說明如何生成一個用於賣場授權的HMAC-SHA256的簽名

1. 組成簽名(sign)的基本字串

字串的規則順序為：**partner_id**, **api path**, **timestamp**

api path: /api/v2/shop/auth_partner

10090/api/v2/shop/auth_partner1594897040



2. 計算加密字串(sign)

將基本字串與 **partner key** 透過 HMAC-SHA256 的雜湊演算法 (Hashing algorithms), 計算出的 16 進位字串 (hex encoded string)

```
90c12d3932f3826f0c72242e1ec6492eec9a1298658f41f7a9469664801c4e5a
```

Python Code Demo:

```
import hmac
import time
import requests
import hashlib

timest = int(time.time())
host = "https://partner.shopeemobile.com"
path = "/api/v2/shop/auth_partner"
redirect_url = "https://www.baidu.com/"
partner_id = 80001
partner_key = "test...."
base_string = "%s%s%s"%(partner_id, path, timest)
sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()

###generate api
url = host + path + "?partner_id=%s&timestamp=%s&sign=%s&redirect=%s"%(partner_id, timest, sign, redirect_url)
```

3. 組成授權連結

將授權 URL, partner_id, redirect, timestamp, sign 如以下範例順序組成授權連結

```
https://partner.test-stable.shopeemobile.com/api/v2/shop/auth_partner?partner_id=10090&redirect=https://www.ABCDEFG.com/&timestamp=1594897040&sign=90c12d3932f3826f0c72242e1ec6492eec9a1298658f41f7a9469664801c4e5a
```

4. 授權完成後的轉跳 URL

賣場授權:

```
https://www.ABCDEFG.com/?code=xxxxxxxx&shop_id=xxxxxx
```

主帳號授權:

```
https://www.ABCDEFG.com/?code=xxxxxxxx&main_account_id=xxxxxx
```

授權逾時邏輯

由於時間戳會於 5 分鐘後失效, 因此授權連結也會於 5 分鐘後過期失效。失效後開發者必須重新生成一個新的授權連結。

點擊授權連結後, 賣家必須在 3 分鐘內輸入賣場帳密或主帳號帳密, 否則提交帳密後將會返回錯誤訊息 "operation overtime"。此時賣家必須重新點擊授權連結, 而非重新整理該頁面。



取得code與shop_id/ main_account_id

完成授權後有兩個方式可以取得：

1. 系統會透過郵件將code與shop_id/ main_account_id寄送至開發者的email
請注意：此方式僅適用於正式環境，測試環境不會寄發
2. code與shop_id/ main_account_id會顯示在轉跳URL中

請特別注意：

v2授權方式同時適用於**v1**與**v2**接口，開發者可以盡早轉換成**v2**授權方式以使用升級後的**API**接口

v1與v2接口可以透過URL內容判斷

v1/v2	HTTP Address
v1	https://partner.shopeemobile.com/api/v1/shop/auth_partner
v2	https://partner.test-stable.shopeemobile.com/api/v2/shop/auth_partner

取消賣場授權

取消授權的方式和步驟與建立授權是相同的，唯一差異是取消授權的URL不同(如以下所示)。開發者若有取消授的需求，可依照建立授權的步驟組成取消授權連結供賣家使用。

取消授權URL

所屬環境	HTTP Address
URL	https://partner.shopeemobile.com/api/v2/shop/cancel_auth_partner
Test URL	https://partner.test-stable.shopeemobile.com/api/v2/shop/cancel_auth_partner



v2接口驗證與請求

API通訊協定

除了特定API接口有在文件中說明使用HTTP/FORM, 其餘API接口皆使用HTTP/JSON

請求/呼叫方式

POST與GET

使用參數

公共參數(common parameters)(輸入)與請求參數(request parameters):
部分API須僅需包含公共參數, 而部分API須包含公共參數與請求參數。針對各API接口需要什麼參數, 請務必參考各接口文檔。

僅需公共參數:



Tables of Contents

- Overview
- Test Menu
- Product
- GlobalProduct
- MediaSpace
- Shop
 - get_shop_info**
 - get_profile
 - update_profile
- Merchant
- Order
- Logistics
- FirstMile
- Payment
- Discount
- TopPicks
- ShopCategory
- Returns
- AccountHealth
- Public
- Push
- Special
- Membership

Shop > get_shop_info > Common Parameter

Common Parameter

Surroundings	HTTP Address
URL	https://partner.uat.shopeemobile.com/api/v2/shop/get_shop_info
Test URL	https://partner.uat.shopeemobile.com/api/v2/shop/get_shop_info

Name	Description
method	GET
data definition	https://open.shopee.com/documents?module=63&type=2&id=50

Name	Type	Sample	Description
partner_id	int	1	Partner ID is assigned upon registration is successful. Required for all requests.
timestamp	timestamp	1610000000	This is to indicate the timestamp of the request. Required for all requests. Expires in 5 minutes.
access_token	string	c09222e3fc40 ffb25fc947f73 8b1abf1	The token for API access, using to identify your permission to the api. Valid for multiple use and expires in 4 hours.
shop_id	int	600000	Shopee's unique identifier for a shop. Required param for most APIs.
sign	string	e318d3e9327 19916a9f9ebb 57e2011961b d47abfa54a36 e040d050d89 31596e2	Signature generated by/(depends on different APIs) partner_id, api path, timestamp, access_token, shop_id(or merchant_id) and partner_key via HMAC-SHA256 hashing algorithm.

Request Parameters

Name	Type	Sample	Description
------	------	--------	-------------

需公共參數與請求參數：



Tables of Contents

- Overview
- Test Menu
- Product
- GlobalProduct
- MediaSpace
- Shop
- Merchant
- Order
- Logistics
- FirstMile
- Payment
- Discount
 - add_discount
 - add_discount_item
 - delete_discount
 - delete_discount_item
 - get_discount**
 - get_discount_list
 - update_discount
 - update_discount_item
 - end_discount
- TopPicks
- ShopCategory
- Returns
- AccountHealth
- Public
- Push
- Special
- Membership

Discount > get_discount > Common Parameter

Common Parameter

Surroundings	HTTP Address
URL	https://partner.uat.shopeemobile.com/api/v2/discount/get_discount
Test URL	https://partner.uat.shopeemobile.com/api/v2/discount/get_discount

Name	Description
method	GET
data definition	https://open.shopee.com/documents?module=63&type=2&id=50

Name	Type	Sample	Description
partner_id	int	1	Partner ID is assigned upon registration is successful. Required for all requests.
timestamp	timestamp	1610000000	This is to indicate the timestamp of the request. Required for all requests. Expires in 5 minutes.
access_token	string	c09222e3fc40 ffb25fc947f73 8b1abf1	The token for API access, using to identify your permission to the api. Valid for multiple use and expires in 4 hours.
shop_id	int	600000	Shopee's unique identifier for a shop. Required param for most APIs.
sign	string	e318d3e9327 19916a9f9ebb 57e2011961b d47abfa54a36 e040d050d89 31596e2	Signature generated by (depends on different APIs) partner_id, api path, timestamp, access_token, shop_id (or merchant_id) and partner_key via HMAC-SHA256 hashing algorithm.

Request Parameters

Name	Type	Sample	Description
discount_id Required	int		Shopee's unique identifier for a discount activity.
page_no Required	int		Specifies the page number of data to return in the current call. Starting from 1. If data is more than one page, the page_no can be some entry to start next call.
page_size Required	int		Each result set is returned as a page of entries. Use the "page_size" filters to control the maximum number of entries to retrieve per page (i.e., per call), and the "page_no" to start next call. This integer value is used to specify the maximum number of entries to return in a single "page" of data.



不同的請求方式(GET/POST), 公共參數與請求參數必須放在不同的位置(Query/Body), 如下所示:

參數	GET	POST
公共參數 Common Param	Query eg. https://partner.shopeemobile.com/api/v2/get_discount?partner_id=844248&shop_id=220286625×tamp=1610359475&access_token=xxx&sign=xxx&discount_id=1000013378	Query eg. https://partner.shopeemobile.com/api/v2/add_discount_item?partner_id=844248&shop_id=220286625×tamp=1610359475&access_token=xxx&sign=xxx
請求參數 Request Param	Query eg. https://partner.shopeemobile.com/api/v2/get_discount?partner_id=844248&shop_id=220286625×tamp=1610359475&access_token=xxx&sign=xxx&discount_id=1000013378	Body eg. <pre>{ "discount_id":1000013378, "item_list":[{ "item_id": 100906910, "purchase_limit": 9, "model_list":[{ "model_id": 10000226319, "model_promotion_price": 600 }] }, { "item_id": 100906913, "purchase_limit": 8, "item_promotion_price": 1500 }] }</pre>

公共參數(輸出)

欄位	類型	必要	說明
request_id	string	yes	API接口請求時的辨識值, 可用來追蹤錯誤
error	string	yes	錯誤代碼。若請求成功則此欄位為空
message	string	no	具體錯誤訊息
warning	string	no	當批次請求時, 部分失敗的訊息會於此欄位說明
response	object	no	具體返回的內容



Access Token

開發者有兩種方式取得access token: GetAccesstoken與RefreshAccessToken

GetAccesstoken

請求方式: GET

說明: 透過授權步驟取得code與shop_id/ main_account_id後, 請求GetAccesstokeAPI獲取shop_id, merchant_id, access_token, refresh_token

URL

所屬環境	HTTP Address
URL	https://partner.shopeemobile.com/api/v2/auth/token/get
Test URL	https://partner.test-stable.shopeemobile.com/api/v2/auth/token/get

公共參數

欄位	類型	必要	說明
sign	string	yes	將基本字串(順序: partner_is, api path, timestamp)和partner_key透過HMAC-SHA256加密後生成簽名
partner_id	int	yes	partner_id可以透過進入Console後, 選擇APP, 在APP中則可取得該資訊。partner_id為所有請求時的必要參數
timestamp	int	yes	請求時的時間戳, 也是所有請求的必要參數, 且會於5分鐘後過期失效

請求參數

欄位	類型	必要	說明
code	string	yes	code可在授權成功後的轉跳URL中獲取, 僅能一次性使用, 且於10分鐘後過期失效
partner_id	int	yes	partner_id為所有請求時的必要參數
shop_id	int	no	授權給開發者的賣場shop_id
main_account_id	int	no	授權給開發者的主帳號main_account_id



返回參數

欄位	類型	必要	說明
request_id	string	yes	API接口請求時的辨識值，可用來追蹤錯誤
error	string	yes	錯誤代碼。若請求成功則此欄位為空
refrest_token	string	yes	使用refresh_token獲取新的access_token。可分別使用於不同的shop_id或merchant_id。每個shop_id或merchant_id為一次性使用，會於 30 天後過期失效
access_token	string	yes	access_token為各API接口的公共請求參數，可多次使用，會於 4 小時後過期失效
expire_in	int	yes	access_token過期失效時間(以秒為單位)
message	string	yes	具體錯誤訊息
merchant_id_list	int[]	no	當請求參數包含main_account_id時會返回
shop_id_list	int[]	no	當請求參數包含main_account_id時會返回

Python Code Demo:




```

import time
import json
import hmac
import hashlib

def get_token_shop_level(code, partner_id, partner_key, shop_id):
    timest = int(time.time())
    body = {"code": code, "shop_id": shop_id, "partner_id": partner_id}
    host = "https://partner.shopeemobile.com"
    path = "/api/v2/auth/token/get"
    base_string = "%s%s%s"%(partner_id, path, timest)
    sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()
    url = host + path + "?partner_id=%s&timestamp=%s&sign=%s"%(partner_id, timest, sign)

    headers = { "Content-Type": "application/json"}
    resp = requests.post(url, json=body, headers=headers)
    ret = json.loads(resp.content)
    access_token = ret.get("access_token")
    new_refresh_token = ret.get("refresh_token")
    return access_token, new_refresh_token

def get_token_account_level(code, partner_id, partner_key, main_account_id):
    timest = int(time.time())
    body = {"code": code, "main_account_id": main_account_id, "partner_id": partner_id}
    host = "https://partner.shopeemobile.com"
    path = "/api/v2/auth/token/get"
    base_string = "%s%s%s"%(partner_id, path, timest)
    sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()
    url = host + path + "?partner_id=%s&timestamp=%s&sign=%s"%(partner_id, timest, sign)

    headers = { "Content-Type": "application/json"}
    resp = requests.post(url, json=body, headers=headers)
    ret = json.loads(resp.content)
    access_token = ret.get("access_token")
    new_refresh_token = ret.get("refresh_token")
    return access_token, new_refresh_token

```



RefreshAccessToken

請求方式:POST

說明：當access_token過期失效後，請求RefreshAccessToken API獲取新的access_token

URL

所屬環境	HTTP Address
URL	https://partner.shopeemobile.com/api/v2/auth/access_token/get
Test URL	https://partner.test-stable.shopeemobile.com/api/v2/auth/access_token/get

公共參數

欄位	類型	必要	說明
sign	string	yes	將基本字串(順序:partner_is, api path, timestamp)和partner_key透過HMAC-SHA256加密後生成簽名
partner_id	int	yes	partner_id可以透過進入Console後，選擇APP，在APP中則可取得該資訊。partner_id為所有請求時的必要參數
timestamp	int	yes	請求時的時間戳，也是所有請求的必要參數，且會於5分鐘後過期失效

請求參數

欄位	類型	必要	說明
refresh_token	string	yes	使用refresh_token獲取新的access_token。可分別使用於不同的shop_id或merchant_id。每個shop_id或merchant_id為一次性使用，會於30天後過期失效
partner_id	int	yes	partner_id為所有請求時的必要參數
shop_id	int	no	授權給開發者的賣場shop_id。shop_id與merchant_id必須被個別請求，一次僅能請求其中一個於請求參數中
merchant_id	int	no	授權給開發者的merchant_id，為獨立辨識值。shop_id與merchant_id必須被個別請求，一次僅能請求其中一個於請求參數中



返回參數

欄位	類型	必要	說明
request_id	string	yes	API接口請求時的辨識值，可用來追蹤錯誤
error	string	yes	錯誤代碼。若請求成功則此欄位為空
refrest_token	string	yes	使用新的refresh_token獲取新的access_token。可分別使用於不同的shop_id或merchant_id。每個對象為一次性使用，會於30天後過期失效
access_token	string	yes	新的access_token, access_token為各API接口的公共請求參數, 可多次使用, 會於4小時後過期失效
expire_in	int	yes	access_token過期失效時間(以秒為單位)
partner_id	int	yes	開發者可於console中的APP取得
shop_id	int	no	授權給開發者的賣場id
merchant_id	int	no	授權給開發者的merchant id

Python Code Demo:



```

#!/usr/bin/env python
# encoding: utf-8

import time
import json
import hmac
import hashlib
import requests

def get_access_token_shop_level(shop_id, partner_id, partner_key, refresh_token):
    timest = int(time.time())
    host = "https://partner.shopeemobile.com"
    path = "/api/v2/auth/access_token/get"
    body = {"shop_id": shop_id, "refresh_token": refresh_token}
    base_string = "%s%s%s"%(partner_id, path, timest)
    sign = hmac.new(partner_key, base_string, hashlib.sha256).hexdigest()
    url = host + path + "?partner_id=%s&timestamp=%s&sign=%s"%(partner_id, timest, sign)

    headers = {"Content-Type": "application/json"}
    resp = requests.post(url, json=body, headers=headers)

    ret = json.loads(resp.content)
    access_token = ret.get("access_token")
    new_refresh_token = ret.get("refresh_token")
    return access_token, new_refresh_token

def get_access_token_merchant_level(merchant_id, partner_id, partner_key, refresh_token):
    timest = int(time.time())
    host = "https://partner.shopeemobile.com"
    path = "/api/v2/auth/access_token/get"
    body = {"merchant_id": merchant_id, "refresh_token": refresh_token}
    base_string = "%s%s%s"%(partner_id, path, timest)
    sign = hmac.new(partner_key, base_string, hashlib.sha256).hexdigest()
    url = host + path + "?partner_id=%s&timestamp=%s&sign=%s"%(partner_id, timest, sign)

    headers = {"Content-Type": "application/json"}
    resp = requests.post(url, json=body, headers=headers)

    ret = json.loads(resp.content)
    access_token = ret.get("access_token")
    new_refresh_token = ret.get("refresh_token")
    return access_token, new_refresh_token

```



計算簽名

此部分說明如何計算請求v2接口時，在公共參數中所需的HMAC-SHA256簽名(sign)

基本字串

有三種類型的API接口，其計算簽名時需要不同的公共參數 (公共參數與各API接口請求時一致) 請依照以下**順序規則**組成基本字串：

- Shop APIs:
partner_id, api path, timestamp, access_token, shop_id
- Merchant APIs:
partner_id, api path, timestamp, access_token, merchant_id
- Public APIs:
partner_id, api path, timestamp

(api path 範例: /api/v2/shop/get_shop_info)

計算簽名

將基本字串與**partner key**透過HMAC-SHA256的雜湊演算法(Hashing algorithms), 計算出的16進位字串(hex encoded string)

Python Code Demo:



```

#!/usr/bin/env python
# encoding: utf-8

import hmac
import time
import requests
import hashlib

timest = int(time.time())
host = "https://partner.shopeemobile.com"
access_token = "random string"

partner_id = 80001
partner_key = "test....."

#### call shop level api
shop_id = 209920
base_string = "%s%s%s%s"%(partner_id, path, timest, access_token, shop_id)
sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()
path = "/api/v2/example/shop_level/get"

url = host + path + "?partner_id=%s&shop_id=%s&timestamp=%s&access_token=%s&sign=%s"%(partner_id, shop_id, timest, access_token, sign)
headers = { "Content-Type": "application/json"}
resp = requests.post(url, headers=headers)

#### call merchant level api
merchant_id = 1234567
base_string = "%s%s%s%s"%(partner_id, path, timest, access_token, merchant_id)
sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()
path = "/api/v2/example/merchant_level/get"

url = host + path + "?partner_id=%s&merchant_id=%s&timestamp=%s&access_token=%s&sign=%s"%(partner_id, merchant_id, timest, access_token, sign)
headers = { "Content-Type": "application/json"}
resp = requests.get(url, headers=headers)

#### call public api
base_string = "%s%s%s%s"%(partner_id, path, timest, access_token)
sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()
path = "/api/v2/auth_merchant/access_token/get"

url = host + path + "?partner_id=%s&timestamp=%s&sign=%s"%(partner_id, timest, sign)
body = {"partner_id":partner_id, "merchant_id": merchant_id,"refresh_token":"testingtoken"}
headers = { "Content-Type": "application/json"}
resp = requests.post(url, json=body, headers=headers)

```

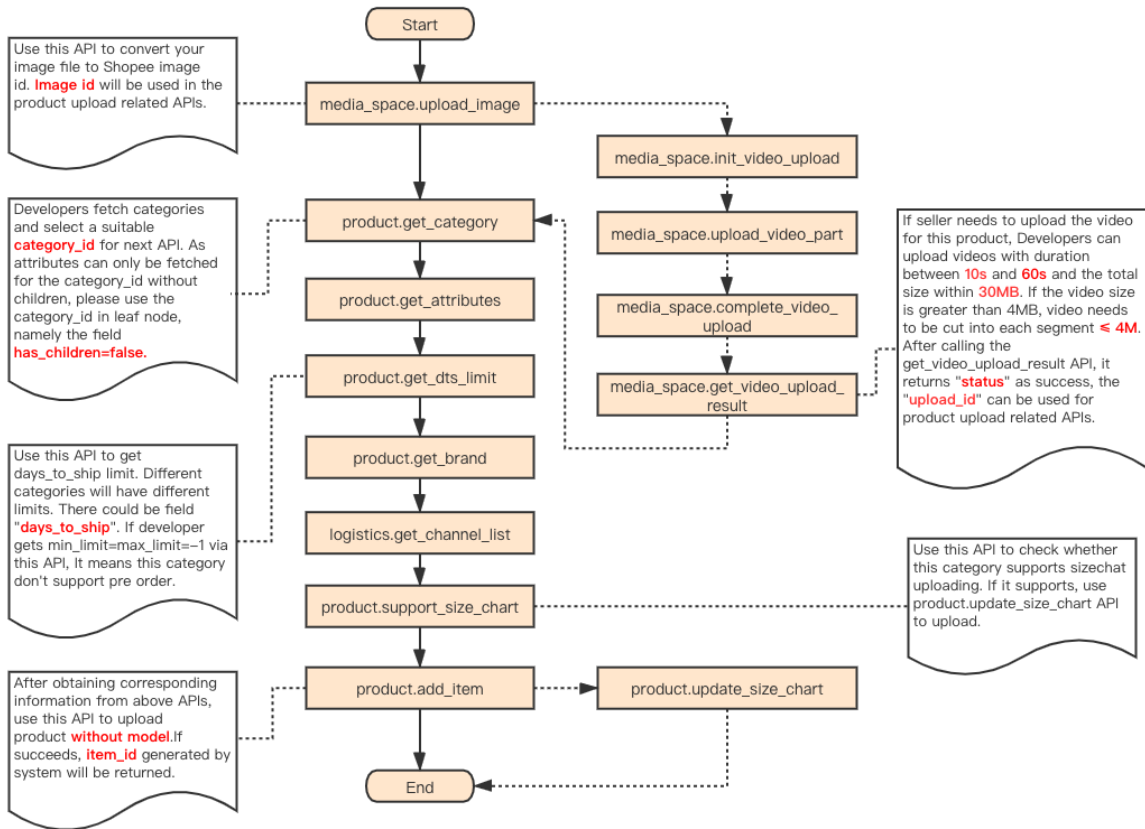


v2核心接口請求流程

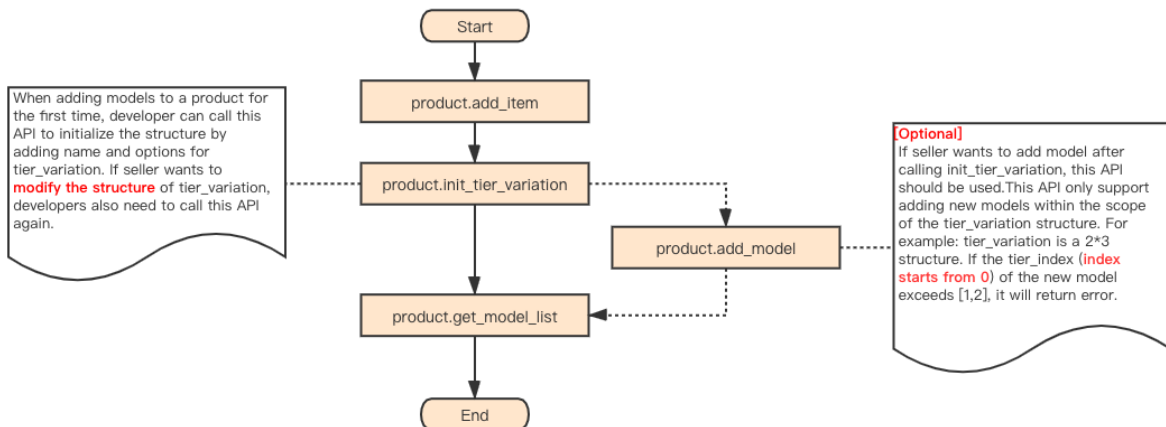
商品上架



1.Add item



2.Add Model



訂單管理與出貨

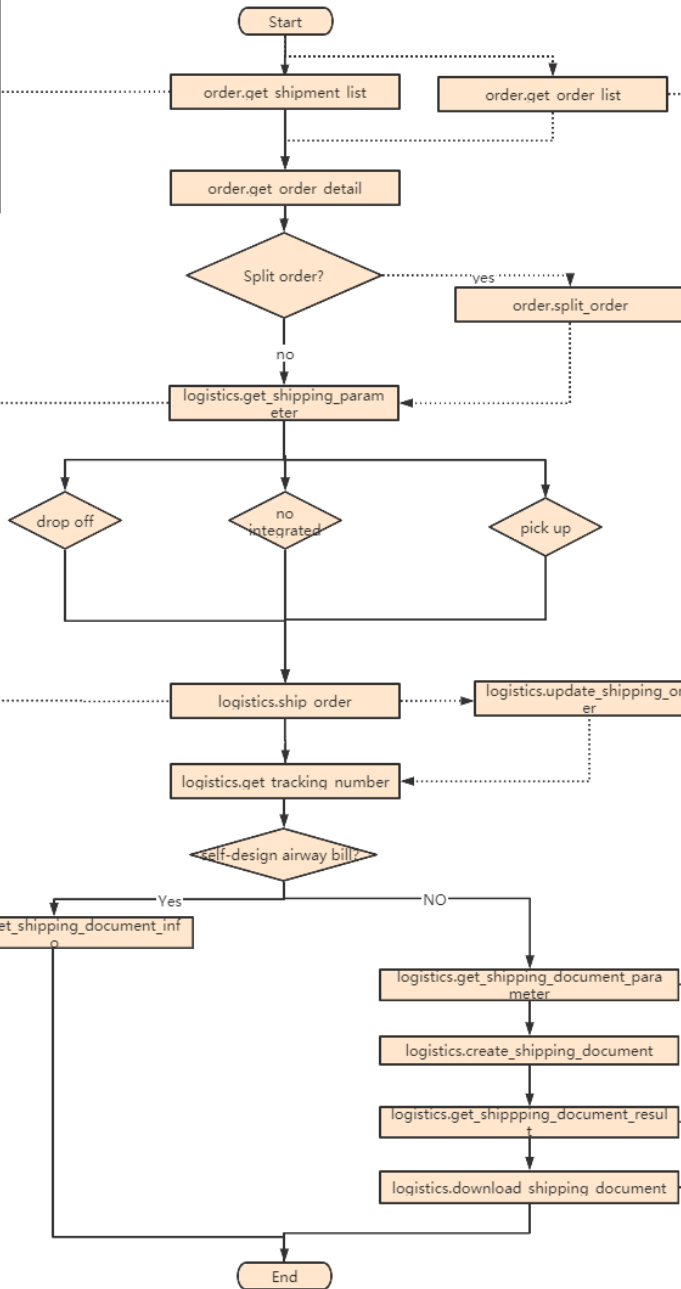
Arrange Shipment

This API returns the order that status is "READY_TO_SHIP" so developer can directly get the list of orders that need to be shipped by calling the order.get_shipment_list API. If the order has been split, some of the order status is already "SHIPPED", but some of the order status is still "READY_TO_SHIP". get_shipment_list API will return the orders and package number which status is "READY_TO_SHIP", but "SHIPPED" package number will not return.

Order logistics channel mode will be reflected from this API:Shopee integrated(pickup & dropoff) and Non-Integrated. Only the returned one can be used to ship. Select **ONLY ONE** of them if multiple modes returned. Get the required input parameters for next API for each mode orders.

ship_order(Arrange Shipment). TrackingNo will not be generated and returned after the call is successful for integrated(pickup & dropoff) channel orders. Developer need to call get_tracking_number API to get TrackingNO. For non-integrated channel order, developer should have the trackingNo ready and inputs in the request body. Integrated channel order status will be automatically updated from **READY TO SHIP** to **PROCESSED**. Non-integrated channel order status will be updated immediately after Init is successful.

Ask your pertinent Shopee business account manager or operation team for the specific logistics channel **airway bill template**, and fetch the necessary field value from this API.



Developer can also filter the "READY_TO_SHIP" orders by calling this API, instead of calling get shipment list API.

If order used "pickup" mode to shipped, Developer can use this api to update pickup address and pickup time for shipping order.

Developers can use this api to get the selectable shipping document type and suggested shipping document type. If you don't select any shipping document type, Shopee will use this as default shipping document type.

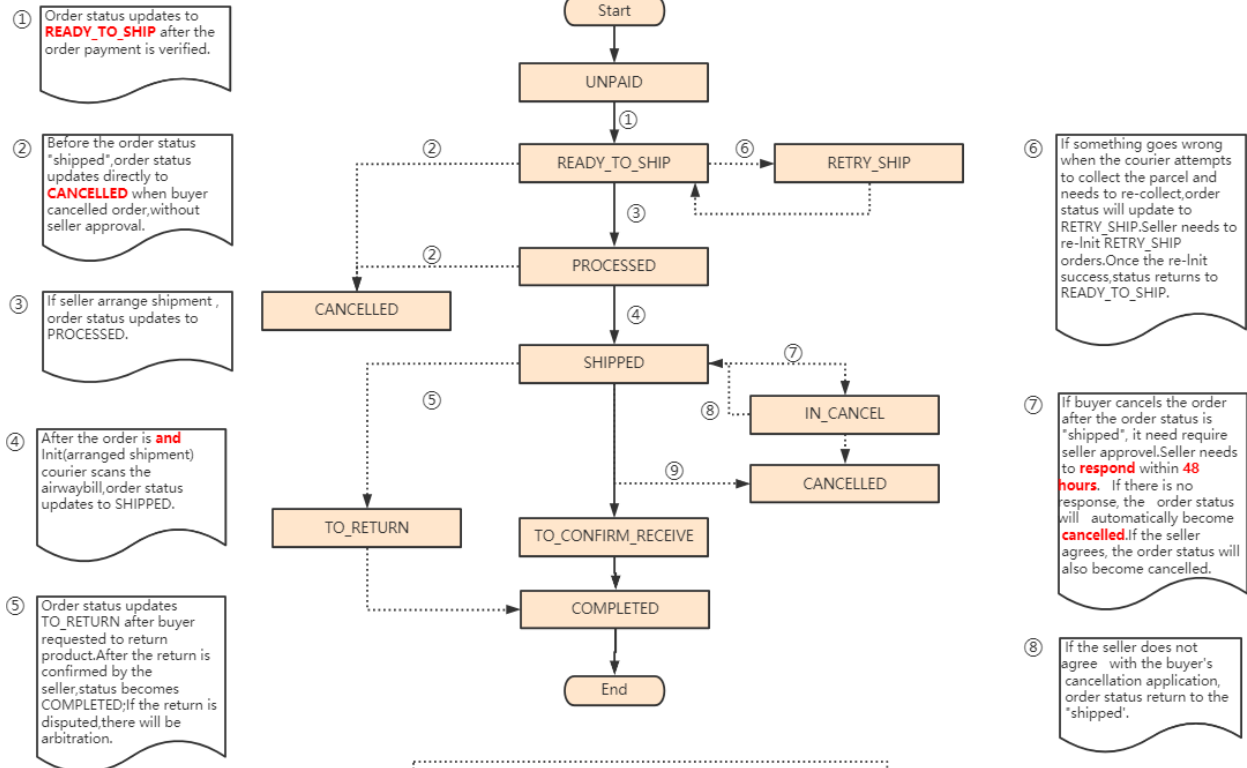
If the task return all order status are "READY", you can download this shipping document success by calling download_shipping_document API. If the task returns part of the order status is "READY" and part of the order status is "FAILED" or "PROCESSED", download_shipping_document API will return failed.

Fetch and print the **AirwayBill**(shipping label). Shopee provides AWB service for most integrated channels and some non-integrated channels. AWB can be fetched only after TrackingNo is generated for most channels. Developer **can no longer** fetch AirwayBill once the order status is updated to **SHIPPED** (indicating the label has been scanned by the couriers).



訂單狀態

Order Status Flow



- ① Order status updates to **READY_TO_SHIP** after the order payment is verified.
- ② Before the order status "shipped", order status updates directly to **CANCELLED** when buyer cancelled order, without seller approval.
- ③ If seller arrange shipment , order status updates to **PROCESSED**.
- ④ After the order is **and** Init(arranged shipment) courier scans the airwaybill, order status updates to **SHIPPED**.
- ⑤ Order status updates **TO_RETURN** after buyer requested to return product.After the return is confirmed by the seller,status becomes **COMPLETED**;If the return is disputed,there will be arbitration.

- ⑥ If something goes wrong when the courier attempts to collect the parcel and needs to re-collect,order status will update to **RETRY_SHIP**.Seller needs to re-Init **RETRY_SHIP** orders.Once the re-Init success,status returns to **READY_TO_SHIP**.
- ⑦ If buyer cancels the order after the order status is "shipped", it need require seller approval.Seller needs to **respond** within **48 hours**. If there is no response, the order status will automatically become **cancelled**.If the seller agrees, the order status will also become cancelled.
- ⑧ If the seller does not agree with the buyer's cancellation application, order status return to the "shipped".
- ⑨ If the package is lost during shopee logistics delivery, the order status will automatically change to cancelled.

Note:

- 1) The number is just a reference,NOT the sequence of the order flow.
- 2)Dotted line indicates the flow is additional

Corresponding Seller Centre Status

OpenAPI	Seller Centre
UNPAID	Unpaid
READY_TO_SHIP	To ship (To ship)
RETRY_SHIP	To ship (To ship)
IN_CANCEL	Cancellation (Cancellation Requested) Shipped (Cancellation Requested)
CANCELLED	Cancellation(Cancelled)
PROCESSED	To ship (Processed)
SHIPPED	Shipping (Shipped)
TO_RETURN	Return/Refund (Return Pending) Shipping (Request Return/Refund)
TO_CONFIRM_RECEIVE	Shipping(Shipped)
COMPLETED	Completed

