



綠界科技股份有限公司

Green World FinTech Service Co., Ltd.

站內付 2.0

新型閘道技術文件_App 版

V 1.0.3

文件編號 gw_p703

2023-09-08

目錄

1. 簡介.....	1
2. 綠界站內付 2.0 流程說明	2
3. 系統介接測試相關資訊.....	10
4. Apple Pay 開發者前置準備說明.....	11
5. 取得廠商驗證碼(Server).....	19
● 交易付款功能.....	19
● 刪除信用卡	27
6. 站內付 2.0 Android SDK(App)	31
7. 站內付 2.0 iOS SDK(App).....	35
8. 付款結果通知.....	40
9. 查詢訂單(Server).....	47
10. 查詢信用卡單筆明細記錄(Server).....	52
11. 信用卡請退款功能(Server).....	56
12. 定期定額查詢(Server).....	61
13. 查詢 ATM/CVS/BARCODE 取號結果(Server)	66
14. 信用卡定期定額訂單作業(Server).....	70
附錄 1. 參數加密方式說明.....	73
附錄 2. 交易訊息代碼一覽表.....	74

附錄 3.	銀行代碼表.....	75
附錄 4.	Enum 列舉說明	76

Version History :

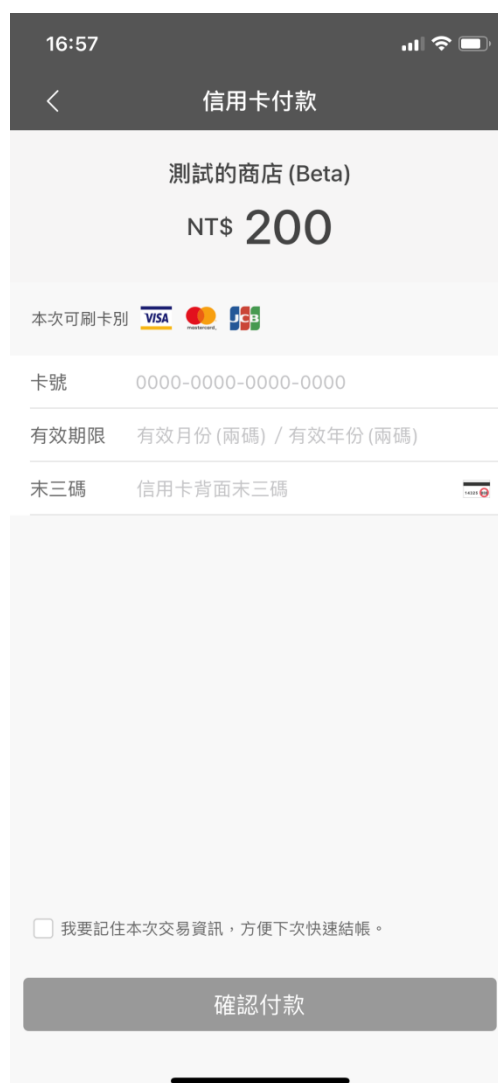
Version	Date	Content
V1.0.0	2022/7/5	Create
V1.0.1	2023/1/3	調整交易金額參數說明
V1.0.2	2023/8/31	調整文件標題
V1.0.3	2023/9/8	特約會員調整為特約賣家

1. 簡介

綠界站內付 2.0 服務對於有收款需求的廠商，提供完整的交易介接技術與多元的付款方式。其中包含信用卡、ATM、超商代碼與超商條碼可供選擇。

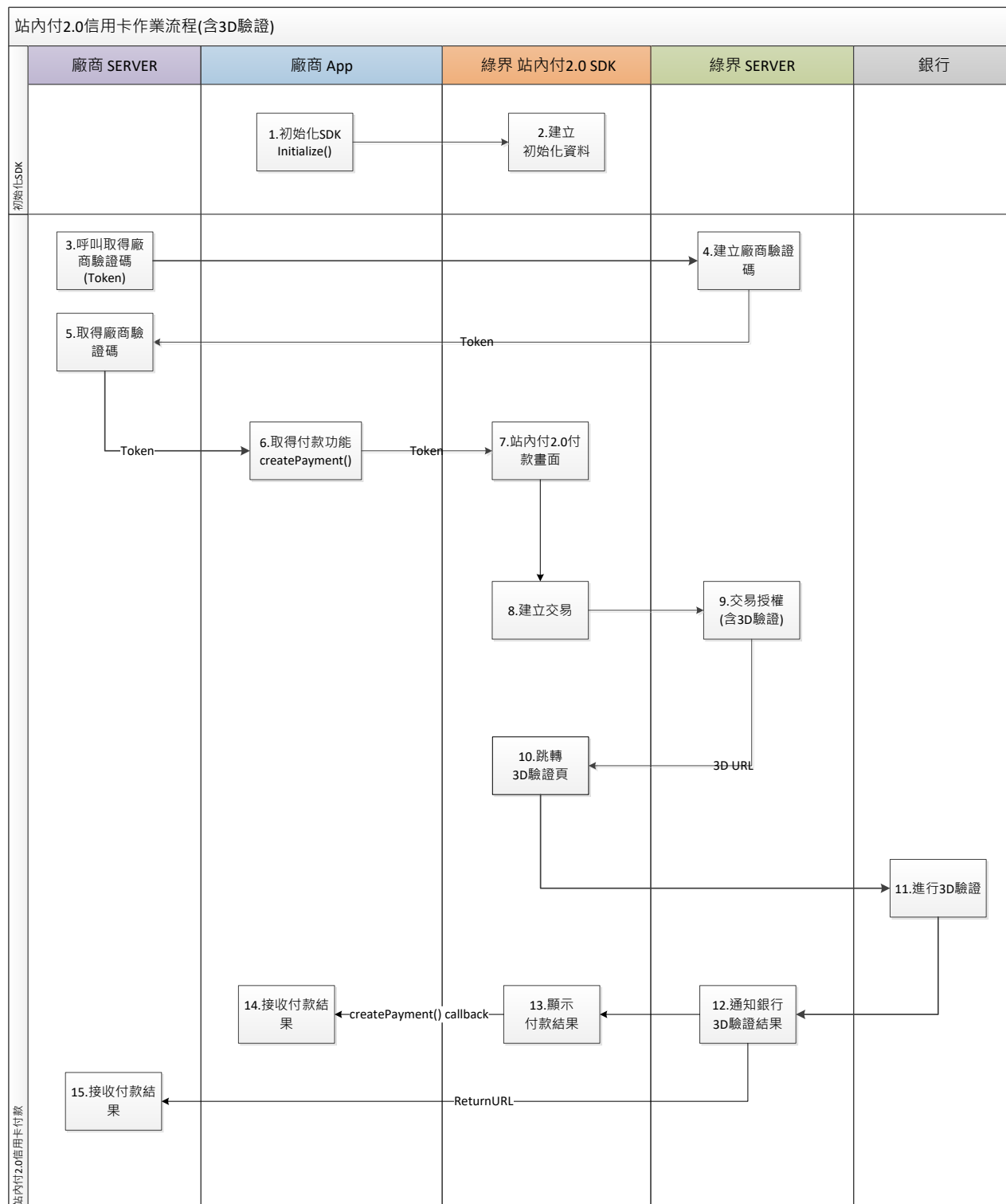
如欲使用站內付 2.0，您必須成為綠界[特約賣家](#)並且申請開通。請前往以下網址並填寫資料，我們將派專人與您聯繫。

<https://member.ecpay.com.tw/MemberReg/MerchantRegister>



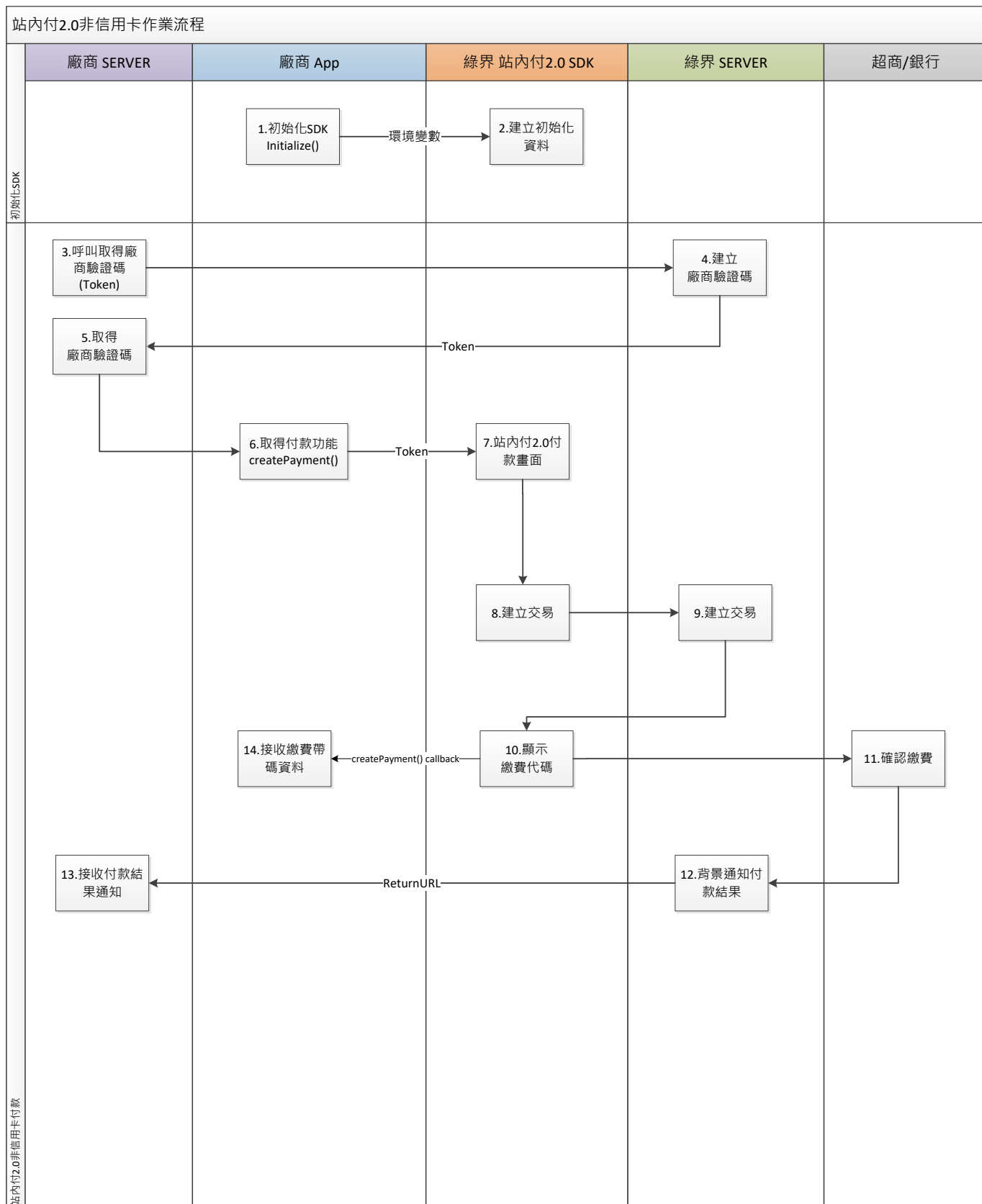
2. 綠界站內付 2.0 流程說明

(1) 站內付 2.0 信用卡作業流程(含 3D 驗證):



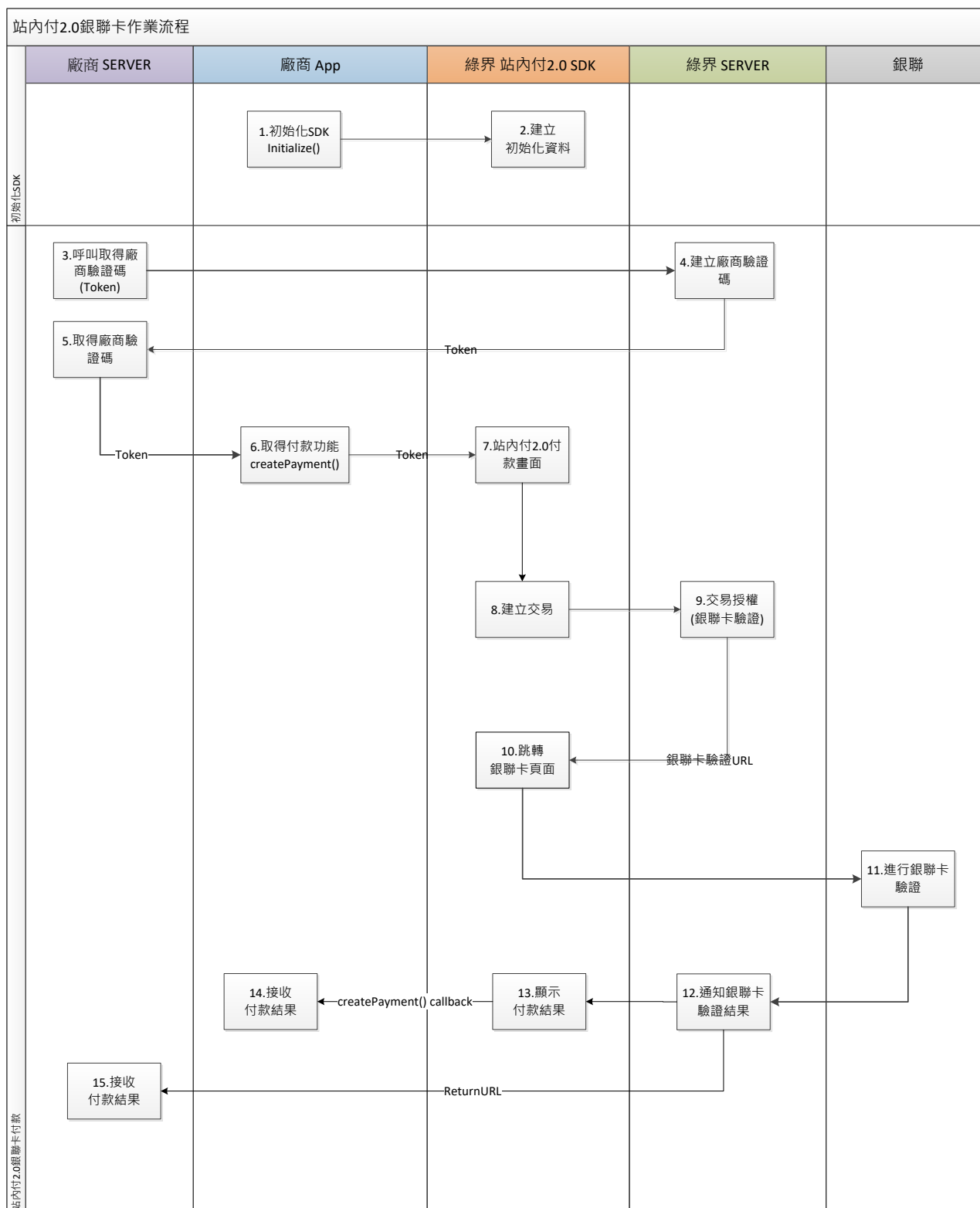
處理角色	流程名稱	處理說明
廠商 App	1.初始化 SDK	開始使用 SDK 需先初始化，並帶入環境變數
綠界站內付 2.0 SDK	2.建立 SDK 初始化資料	SDK 建立初始化資料
廠商 Server	3.呼叫取得 廠商驗證碼	廠商確定訂單內容後，傳送訂單資料到綠界並取得廠商驗證碼(Token)
綠界 Server	4.建立廠商驗證碼	綠界 Server 接收到訂單資料後，產生一組廠商驗證碼(Token)
廠商 Server	5.取得廠商驗證碼	接收綠界回傳的廠商驗證碼(Token)
廠商 App	6.取得付款功能	呼叫取得付款功能
綠界站內付 2.0 SDK	7.站內付 2.0 付款畫面	產生付款畫面
綠界站內付 2.0 SDK	8. 建立交易	將交易資料送至綠界 Server 建立訂單
綠界 Server	9.交易授權(含 3D 驗證)	將信用卡資料送至銀行進行授權與 3D 驗證
綠界站內付 2.0 SDK	10.跳轉 3D 驗證頁	跳轉至 3D 驗證畫面
銀行	11.進行 3D 驗證	銀行進行 3D 驗證
綠界 Server	12.通知銀行 3D 驗證結果	接收 3D 驗證結果
綠界站內付 2.0 SDK	13.顯示付款結果	顯示付款結果
廠商 App	14.接收付款結果	廠商接收 createPayment()回傳的付款結果
廠商 Server	15.接收付款結果	接收 ReturnURL 回傳的付款結果

(2) 站內付 2.0 非信用卡作業流程:



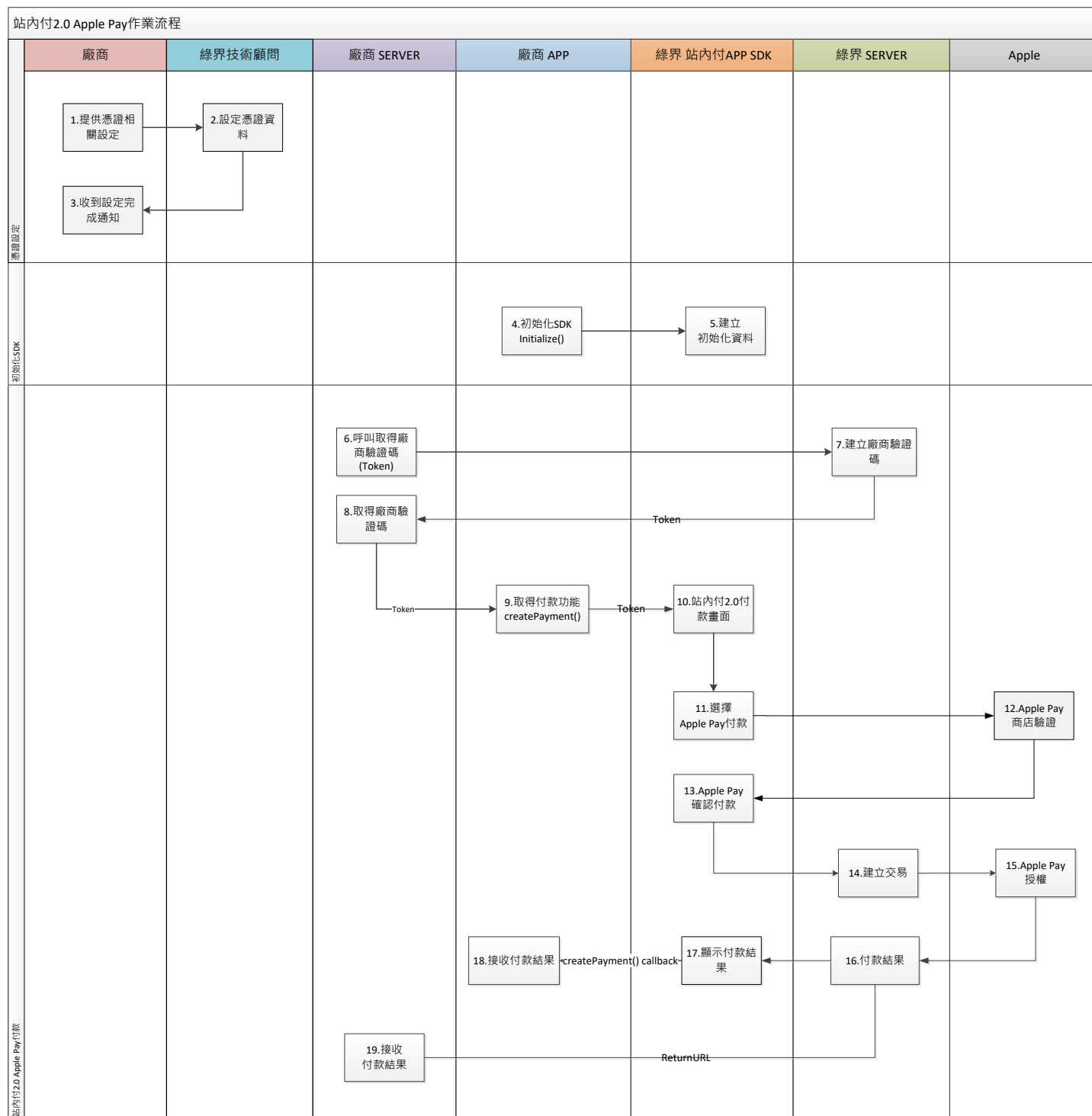
處理角色	流程名稱	處理說明
廠商 App	1.初始化 SDK	開始使用 SDK 需先初始化，並帶入環境變數
綠界站內付 2.0 SDK	2.建立 SDK 初始化資料	SDK 建立初始化資料
廠商 Server	3.呼叫取得 廠商驗證碼	廠商確定訂單內容後，傳送訂單資料到綠界並取得廠商驗證碼
綠界 Server	4.建立廠商驗證碼	綠界 Server 接收到訂單資料後，產生一組廠商驗證碼
廠商 Server	5.取得廠商驗證碼	接收綠界回傳的廠商驗證碼
廠商 App	6.取得付款功能	呼叫取得付款功能
綠界站內付 2.0 SDK	7.站內付 2.0 付款畫面	產生付款畫面
綠界站內付 2.0 SDK	8.建立交易	將交易資料送至綠界 Server 建立訂單並取得取號結果
廠商 Server	9.取得取號結果	取得綠界回傳取號結果資料
廠商 App	10.顯示繳費代碼	顯示繳費代碼提供消費者進行繳費
銀行/超商	11.確認繳費	當消費者完成付款，通知綠界完成繳費
綠界 Server	12.背景通知付款結果	透過 ReturnURL 背景通知廠商付款結果
廠商 Server	13.接收付款結果通知	透過 ReturnURL 背景接收付款結果通知
廠商 App	14.接收繳費代碼資料	廠商接收 createPayment()回傳的繳費代碼資料

(3) 站內付 2.0 銀聯卡作業流程:



處理角色	流程名稱	處理說明
廠商 App	1.初始化 SDK	開始使用 SDK 需先初始化，並帶入環境變數
綠界站內付 2.0 SDK	2.建立 SDK 初始化資料	SDK 建立初始化資料
廠商 Server	3.呼叫取得 廠商驗證碼	廠商確定訂單內容後，傳送訂單資料到綠界並取得廠商驗證碼
綠界 Server	4.建立廠商驗證碼	綠界 Server 接收到訂單資料後，產生一組廠商驗證碼
廠商 Server	5.取得廠商驗證碼	接收綠界回傳的廠商驗證碼
廠商 App	6.取得付款功能	呼叫取得付款功能
綠界站內付 2.0 SDK	7.站內付 2.0 付款畫面	產生付款畫面
綠界站內付 2.0 SDK	8.建立交易	將交易資料送至綠界 Server 建立訂單
綠界 Server	9.交易授權(銀聯卡驗證)	將信用卡資料送至銀聯取得付款畫面
綠界站內付 2.0 SDK	10.跳轉銀聯卡頁面	跳轉至銀聯卡付款畫面
銀聯	11.進行銀聯卡驗證	銀聯進行信用卡驗證
綠界 Server	12.通知銀聯卡驗證結果	從銀聯取得銀聯卡刷卡結果且通知前端付款結果
綠界站內付 2.0 SDK	13.顯示付款結果	顯示付款結果畫面
廠商 App	14.接收付款結果	廠商接收 createPayment()回傳的付款結果
廠商 Server	15.接收付款結果	透過 ReturnURL 背景接收付款結果通知

(4) 站內付 2.0 Apple Pay 作業流程:



處理角色	流程名稱	處理說明
廠商	1.提供憑證相關設定	在串接 Apple Pay 功能之前，需來信聯絡綠界技術顧問並提供 Apple Pay 憑證相關資料。產生憑證方法請參考 Apple Pay 開發者前置準備說明 。 ※注意事項： Apple Pay 功能不提供測試環境，請於正式環境產生憑證進行測試。
綠界技術顧問	2.設定憑證資料	綠界技術顧問設定憑證相關資料
廠商	3.收到設定完成通知	廠商收到綠界技術顧問回信通知設定結果，設定成功後才能使用 Apple Pay 功能
廠商 App	4.初始化 SDK	開始使用 SDK 需先初始化，並帶入環境變數
綠界站內付 2.0 SDK	5.建立 SDK 初始化資料	SDK 建立初始化資料
廠商 Server	6.呼叫取得 廠商驗證碼	廠商確定訂單內容後，傳送訂單資料到綠界並取得廠商驗證碼(Token)
綠界 Server	7.建立廠商驗證碼	綠界 Server 接收到訂單資料後，產生一組廠商驗證碼(Token)
廠商 Server	8.取得廠商驗證碼	接收綠界回傳的廠商驗證碼(Token)
廠商 App	9.取得付款功能	呼叫取得付款功能
綠界站內付 2.0 SDK	10.站內付 2.0 付款畫面	產生付款畫面
綠界站內付 2.0 SDK	11.選擇 Apple Pay 付款	使用者於付款畫面選擇 Apple Pay 付款
Apple	12. Apple Pay 商店驗證	將商店資料送至 Apple 進行商店驗證
綠界站內付 2.0 SDK	13. Apple Pay 確認付款	付款畫面顯示 Apple Pay 付款按鈕，並按下確認付款
綠界 Server	14. 建立交易	建立交易訂單
Apple	15. Apple Pay 授權	綠界將交易訂單送至 Apple 進行授權
綠界 Server	16.付款結果	收到 Apple 回傳的授權結果
綠界站內付 2.0 SDK	17.顯示付款結果	顯示付款結果畫面
廠商 APP	18.接收付款結果	廠商接收 createPayment()回傳的付款結果
廠商 Server	19.接收付款結果	透過 ReturnURL 接收付款結果

3. 系統介接測試相關資訊

請使用以下資訊介接本公司金流服務，此為測試環境，測試無誤要連正式環境時，請將以下資訊更換成正式環境的相關資訊。

欄位說明	欄位內容	
特店編號(MerchantID)	3002607	--
閘道商特店編號 (GatewayMerchantID)	--	3085779
登入廠商後台帳號	stagetest3	gatewaytest02
登入廠商後台密碼	test1234	
統一編號	53538851	
廠商後台測試環境	https://vendor-stage.ecpay.com.tw 此網站提供查詢測試的訂單相關資訊，也可執行訂單模擬付款的功能，送回付款通知的資訊到貴公司的網站。	
all in one 介接的 HashKey	pwFHCqoQZGmho4w6	y6869NBszTuvhSRx
all in one 介接的 HashIV	EkRm7iFT261dpevs	BMm7FmX91dE8rpdw
信用卡測試卡號	4311-9522-2222-2222(注意事項:只有此組卡號可測試交易成功流程)	
信用卡測試安全碼	222	
信用卡測試有效年月	請設定大於測試時間。假如您的測試時間在 2013 年 11 月 26 號，該筆交易的信用卡有效年月請設定 2013 年 11 月以後，因為系統會判斷有效年月是否已過期，已過期則會回應刷卡失敗。	

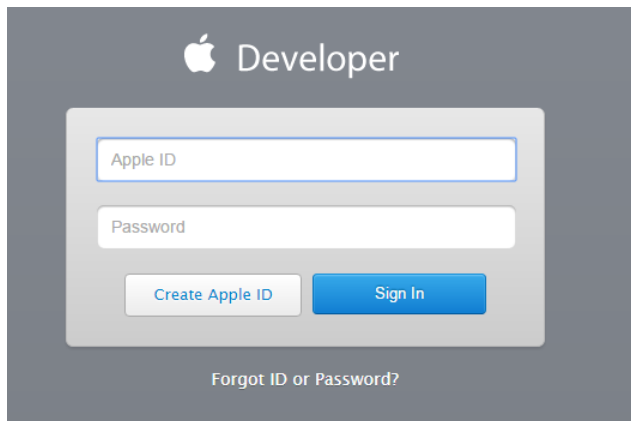
※注意事項：

1. 接收傳送綠界 API 通知時，請開發人員確認下面事項，以利正常收到綠界發送的各项通知：
2. 請確認 Server 是否有開通防火牆，以避免付款通知被防火牆阻擋。
3. 呼叫 API 連接 port 只提供 https (443 port)連線方式，並請使用 DNS(Domain Name System)進行介接。
4. 請確認各項交易參數傳送時是使用 Http POST 方式傳送至綠界 API。
5. 請確認 Server URL 連接 port 為 http 80 port 與 https 443 port。
6. 測試環境信用卡付款方式未提供海外卡及銀聯卡服務。
7. 傳送參數值的內容請用 UTF8 格式，且不支援中英文語系之外的文字內容。
8. 請先申請關閉 OTP 服務及申請開啟信用卡 3D 驗證服務才能完成串接交易。
9. 為保障消費者權益與網路交易安全，串接服務僅支援 TLS 1.1 以上之加密通訊協定。
10. 綠界主機 IP 不固定，如廠商防火牆需開通綠界 IP，請以 FQDN 方式設定以下 domain:
連入:postgate.ecpay.com.tw、postgate-stage.ecpay.com.tw
連出:payment.ecpay.com.tw、payment-stage.ecpay.com.tw
11. Apple Pay 功能不提供測試環境使用，請於正式環境產生憑證進行測試。

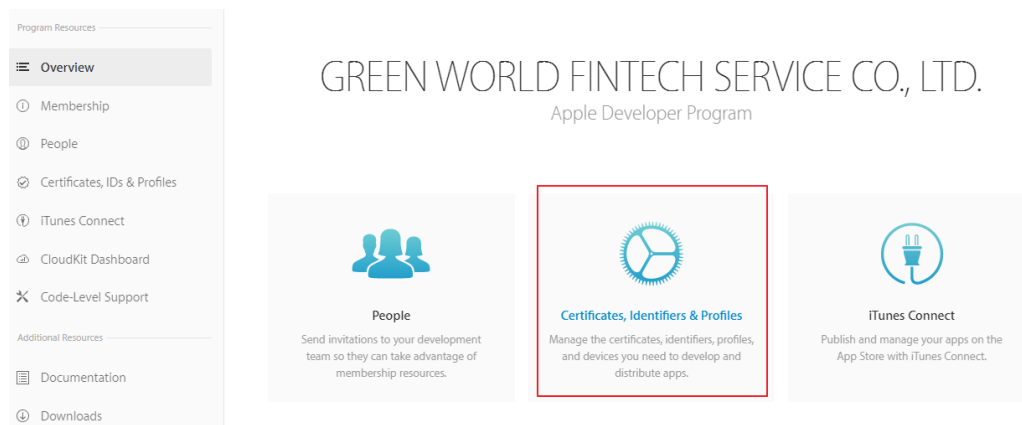
4. Apple Pay 開發者前置準備說明

- 建立商務帳號(MerchantID):

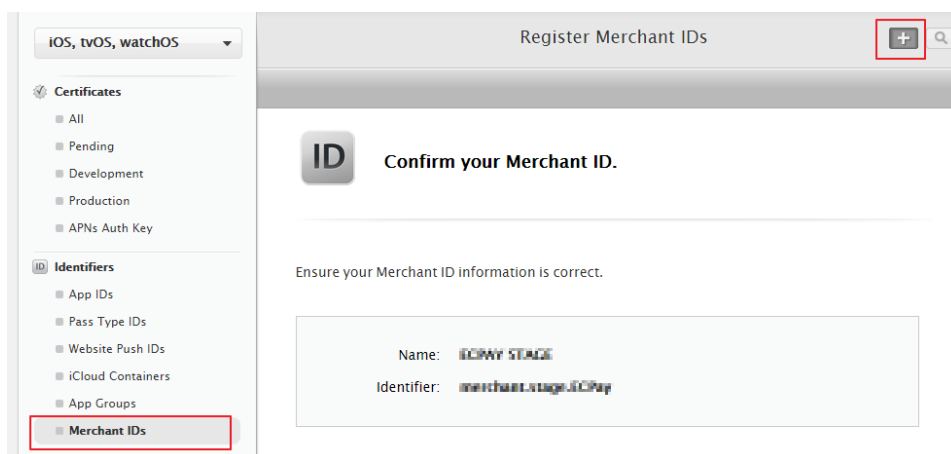
1. 請登入至您的 Apple 開發者帳號



2. 選擇“Certificates, Identifiers & Profiles.”



3. 選擇“Merchant IDs”



4. 註冊商務資料

Register Merchant IDs

ID Registering a Merchant ID

Register your Merchant Identifiers (Merchant IDs) to enable your apps to process transactions for physical goods and services to be used outside of your apps. Generate a Apple Pay certificate for each registered Merchant ID to validate transactions initiated within your app.

Merchant ID Description

Description:
You cannot use special characters such as @, &, %, ', ", "

Identifier
Enter a unique identifier for your Merchant ID, starting with the string 'merchant'.

ID:
We recommend using a reverse-domain name style string (i.e., merchant.com.example.merchantname).

5. 確認註冊商務資料

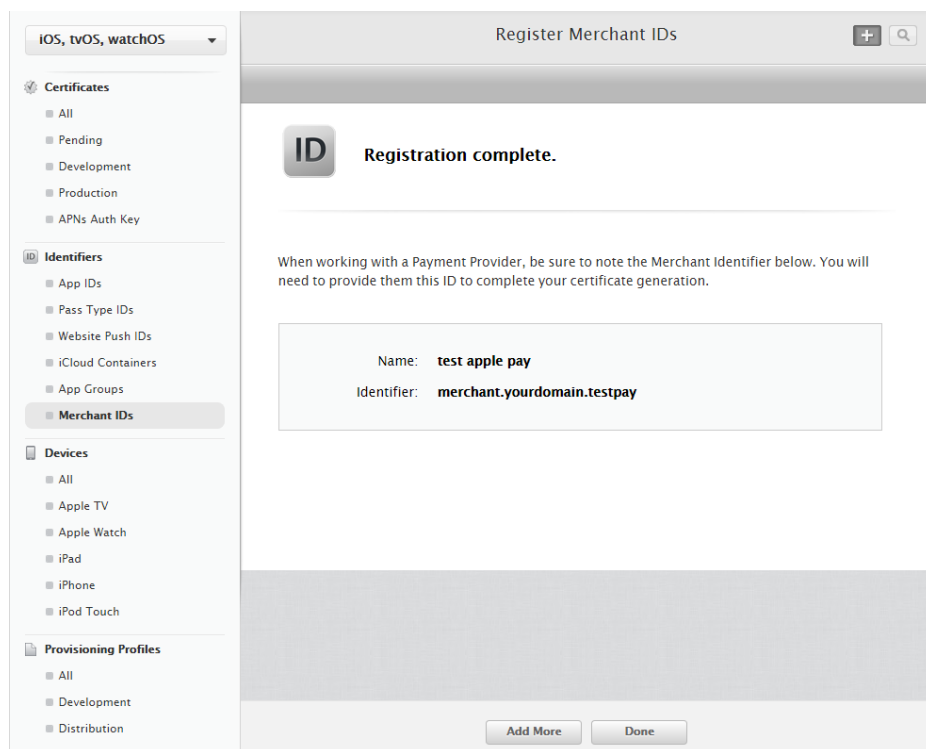
Register Merchant IDs

ID Confirm your Merchant ID.

Ensure your Merchant ID information is correct.

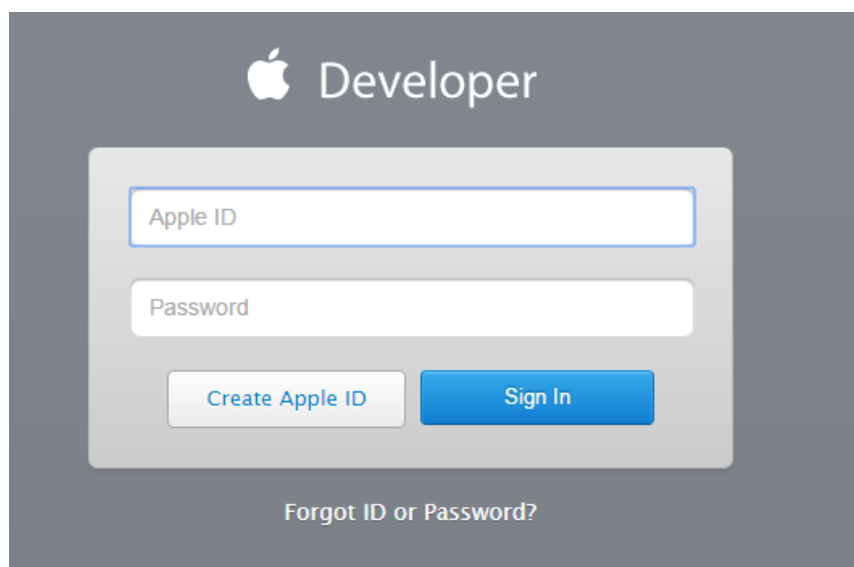
Name: **test apple pay**
Identifier: **merchant.yourdomain.testpay**

6. 完成註冊商務資料

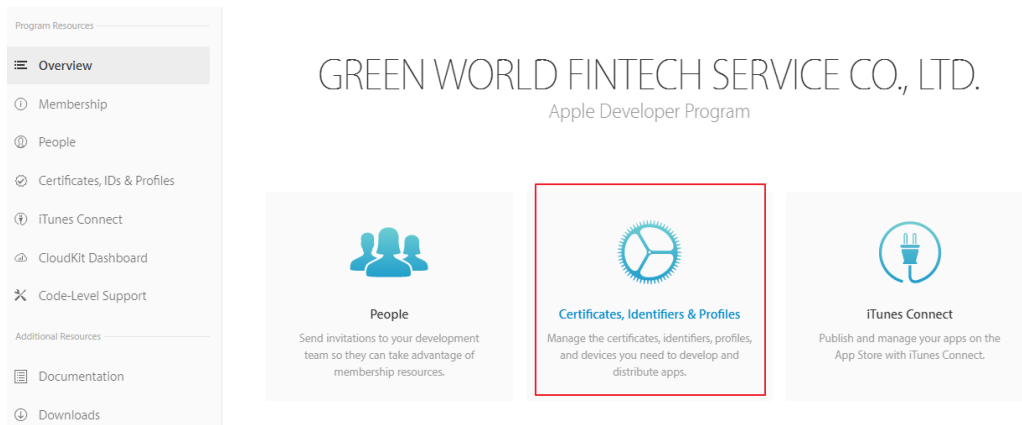


● 建立交易憑證(Payment Processing Certificate):

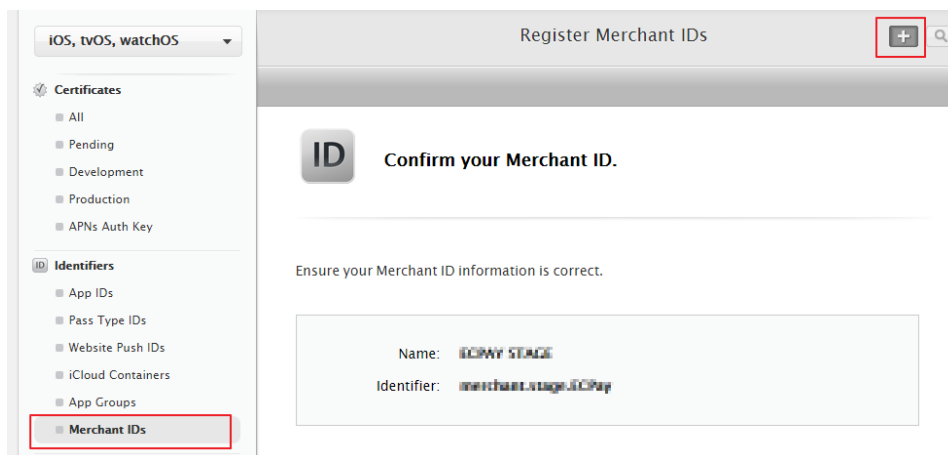
1. 請登入至您的 Apple 開發者帳號



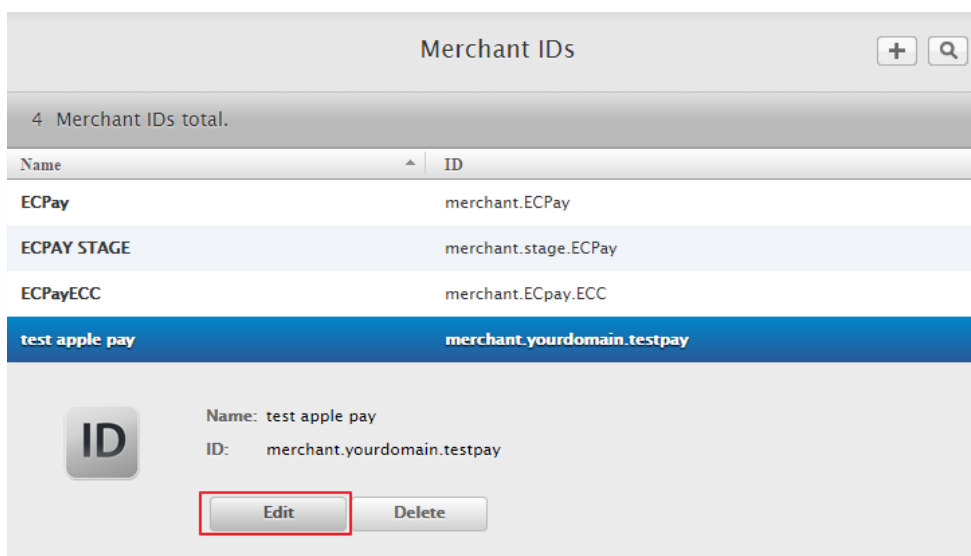
2. 選擇“Certificates, Identifiers & Profiles.”



3. 選擇 Merchant IDs



4. 選擇要新增憑證的 Merchant ID 並且按下 Edit



5. 選擇 Payment Processing Certificate 裡的 Create Certificate 按鈕，建立商務交易憑證

Setup and configure services for this Merchant ID.

ID

ID: merchant.yourdomain.testpay
Name: test apple pay

To configure Apple Pay for this merchant ID, create a Payment Processing Certificate. Apple Pay requires this certificate to encrypt transaction data. Use the same certificate for Apple Pay in apps or on the web.

Payment Processing Certificate

Create a payment processing certificate for this merchant ID.

Create Certificate

6. 回答問題,請選擇 NO 並按 Continue

Setup and configure services for this Merchant ID.

ID

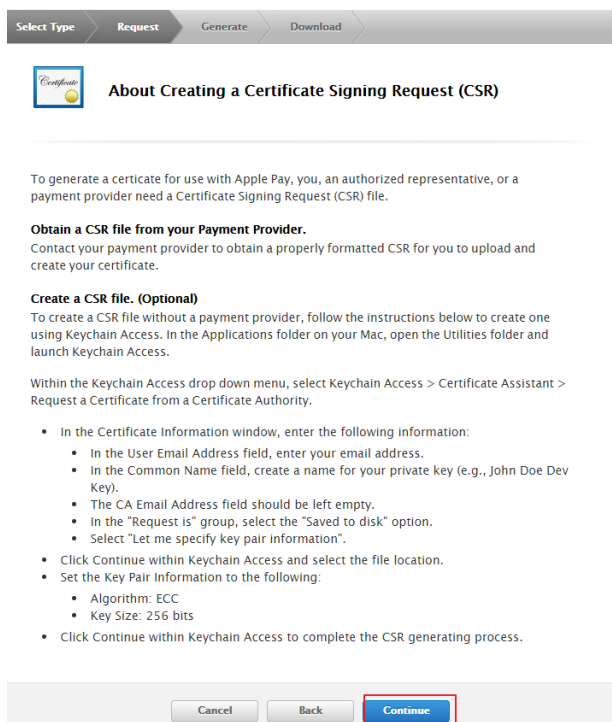
ID: merchant.yourdomain.testpay
Name: test apple pay

Will payments associated with this Merchant ID be processed exclusively in China?

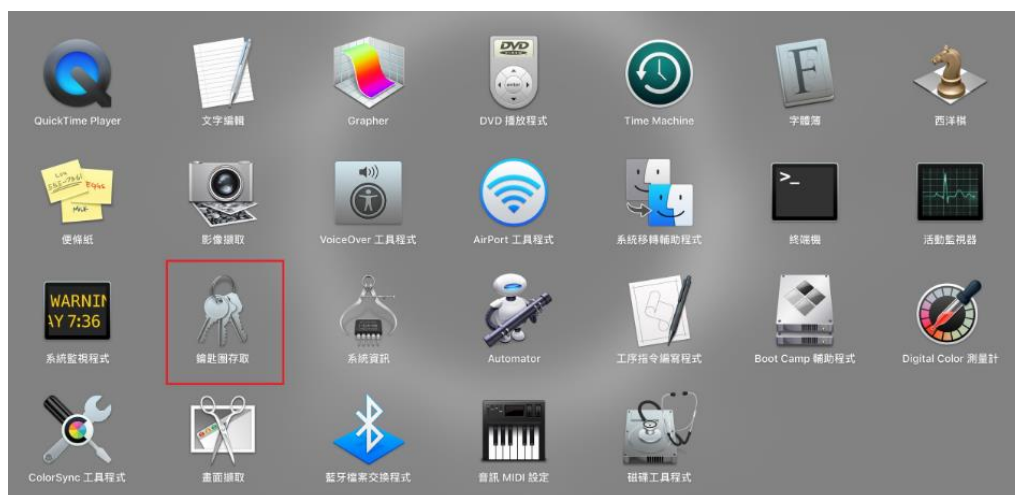
☒ No
☐ Yes

Continue

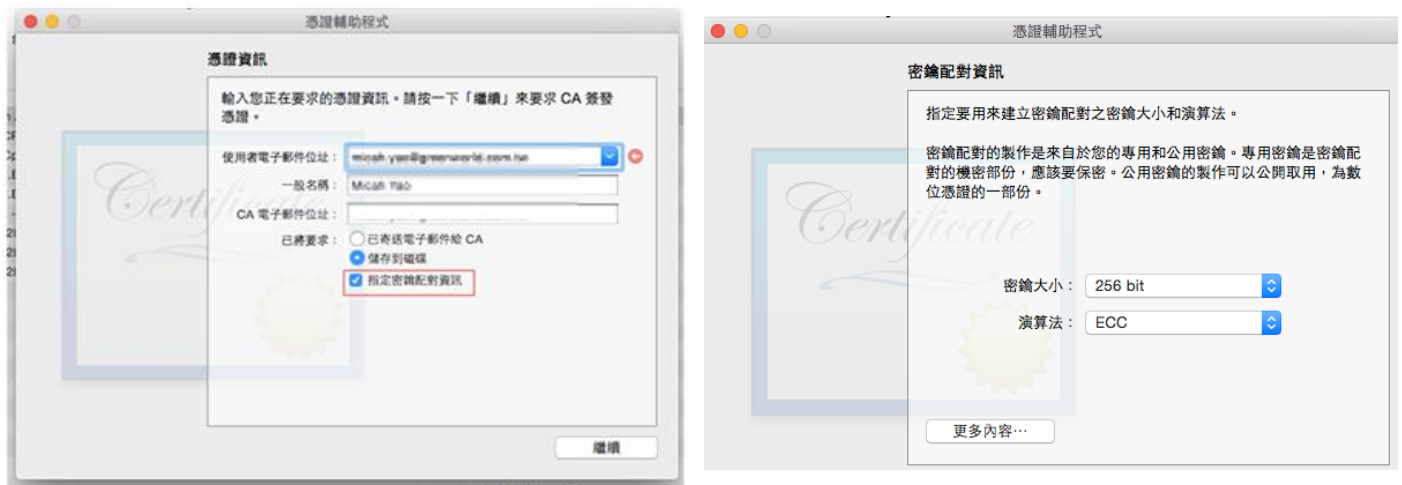
7. 閱讀憑證請求檔建立相關資訊後,按 Continue



8. 請使用 mac 電腦建置憑證請求檔

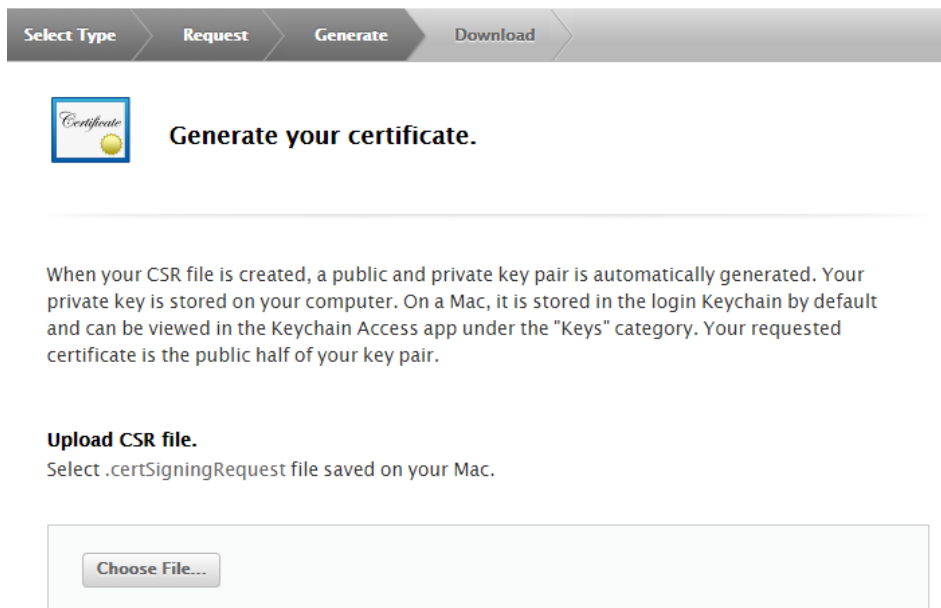


9. 選擇憑證演算法及長度(演算法請選擇 ECC,長度請選擇 256) · 並選擇儲存到磁碟



10. 將檔案直接儲存到硬碟後，取得憑證請求檔 CSR

11. 上傳憑證請求檔



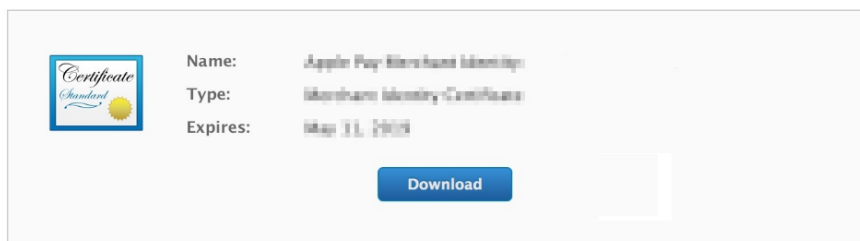
12. 點擊 Download 按鈕來下載憑證，將它安裝至 mac 電腦鑰匙圈存取



Your certificate is ready.

Download, Install and Backup

Download your certificate to your Mac, then double click the .cer file to install in Keychain Access. Make sure to save a backup copy of your private and public keys somewhere secure.



13. 選擇交易憑證，匯出檔案 p12 格式(必須設定憑證密碼)



14. 把匯出後的憑證檔案 (p12 格式) 及憑證密碼提供給綠界科技

5. 取得廠商驗證碼(Server)

● 交易付款功能

- 應用場景：在使用綠界站內付 2.0 金流服務之前，廠商 Server 必須先向綠界 Server 取得一組廠商驗證碼(Token)。廠商 Server 得到驗證碼後，必須將驗證碼傳給廠商 App 做產生站內付 2.0 金流畫面功能之用。
- 介接路徑
 - 正式環境：<https://ecpg.ecpay.com.tw/Merchant/GetTokenbyTrade>
 - 測試環境：<https://ecpg-stage.ecpay.com.tw/Merchant/GetTokenbyTrade>
- 廠商傳入參數說明：(JSON 格式，名稱若有紅色星號*為必填欄位)
 - Content Type : application/json
 - HTTP Method : POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
	*Timestamp	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
	*Revision	String(10)	串接文件版號 請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{  
  "MerchantID": "3002607",  
  "RqHeader": {  
    "Timestamp": 1525168923,  
    "Revision": "1.0.0"  
  },  
  "Data": "..."}  
}
```

● Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	特約合作平台商代號	String(10)	為專案合作的平台商使用。一般特店或平台商本身介接，則參數請帶空值。若為專案合作平台商的特店使用時，則參數請帶平台商所綁的特店編號 [MerchantID]。
*GatewayMerchantID	閘道商特店編號	String(10)	實際建立訂單的金流模式，會依照廠商後台的閘道介接設定內容為主。
*RememberCard	是否使用記憶卡號功能	Int	此功能會將消費者的信用卡記憶在 GatewayMerchantID 上 0:否 1:是
*PaymentUIType	畫面的呈現方式	Int	0: 信用卡定期定額 1: 國民旅遊卡 2: 付款選擇清單頁
ChoosePaymentList	欲使用的付款方式	String(30)	PaymentUIType 如選擇 2，則必填 0: 全部付款方式 1: 信用卡一次付清 2: 信用卡分期付款 3: ATM 4: 超商代碼 5: 超商條碼 6: 銀聯卡 7: Apple Pay (憑證設定相關流程請參考產生憑證方法請參考 Apple Pay 開發者前置準備說明) 可多選 ex: 1,2,3
*OrderInfo	訂單資訊	JSON	
	*MerchantTradeDate	String(20)	廠商交易時間 yyyy/MM/dd HH:mm:ss
	*MerchantTradeNo	String(20)	均為唯一值，不可重複使用。英數字大小寫混和 如何避免訂單編號重複請參考 FAQ 。
	*TotalAmount	Int	請帶整數，不可有小數點。 僅限新台幣，且不可超過 20 萬。其餘各付款金額的限制，請參考： https://www.ecpay.com.tw/CascadeFAQ/CascadeFAQ_Qa?nID=3605
	*ReturnURL	String(200)	當消費者付款完成後，綠界會將付款結果參數以幕後(Server POST)回傳到該網址。詳細說明請參考付款結果通知 注意事項： 1. 請勿設定與 Client 端接收付款結果網址 OrderResultURL 相同位置，避免程式判斷錯誤。

				2. 請在收到 Server 端付款結果通知後，請正確回應 1 OK 給綠界。 3. 1 OK 僅是廠商回應綠界是否收到通知，並不會改變付款狀態。
	*TradeDesc	交易描述	String(200)	
	*ItemName	商品名稱	String(400)	商品名稱以分隔線#分開
CardInfo	信用卡資訊	JSON		以下情況為必填： 1. PaymentUIType 選擇 0 或 1 2. PaymentUIType 選擇 2，且 ChoosePaymentList 選擇 0、1 或 2
	Redeem	使用信用卡紅利	String(1)	預設為不使用 0: 不使用 1: 使用
	PeriodAmount	定期定額每次授權金額	Int	當 PaymentUIType 為 0 時，此欄位必填
	PeriodType	定期定額週期種類	String(1)	當 PaymentUIType 為 0 時，此欄位必填 D：以天為週期 M：以月為週期 Y：以年為週期
	Frequency	定期定額執行頻率	Int	當 PaymentUIType 為 0 時，此欄位必填 注意事項： 至少要大於等於 1 次以上。 當 PeriodType 設為 D 時，最多可設 1~365(天)。 當 PeriodType 設為 M 時，最多可設 1~12(月)。 當 PeriodType 設為 Y 時，最多可設 1。
	ExecTimes	定期定額執行次數	Int	當 PaymentUIType 為 0 時，此欄位必填。 注意事項： 至少要大於 1 次以上。 當 PeriodType 設為 D 時，最多可設 999 次。 當 PeriodType 設為 M 時，最多可設 99 次。 當 PeriodType 設為 Y 時，最多可設 9 次。
	*OrderResultURL	3D 驗證回傳付款結果網址	String(200)	使用 APP SDK 時，3D 驗證結果將於 SDK callback 回傳。 此欄位於 APP SDK 目前無作用，請固定填入 https://www.ecpay.com.tw/ 即可
	PeriodReturnURL	定期定額執行結果回應 URL	String(200)	當 PaymentUIType 為 0 時，此欄位必填 若交易是信用卡定期定額的方式，則每次執行授權完，會將授權結果回傳到這個設定的 URL。
	CreditInstallment	刷卡分期期數	String(20)	當 ChoosePaymentList 為 0 或 ChoosePaymentList 有選擇 2 時，此欄位必填 支援多期數請以逗號分隔 ex: 3,6,12,18,24

	TravelStartDate	國旅卡旅遊起 日	String(8)	當 PaymentUIType 為 1 時，此欄位必填 格式為 mmddyyyy
	TravelEndDate	國旅卡旅遊訖 日	String(8)	當 PaymentUIType 為 1 時，此欄位必填 格式為 mmddyyyy
	TravelCounty	國旅卡旅遊縣 市	String(3)	當 PaymentUIType 為 1 時，此欄位必填 縣市群組名稱 縣市群組代碼 台北市 001 基隆市 002 新北市 003 宜蘭縣 004 新竹市 005 新竹縣 006 桃園縣 007 苗栗縣 008 台中市 009 彰化縣 011 南投縣 012 嘉義市 013 嘉義縣 014 雲林縣 015 台南市 016 高雄市 018 屏東縣 020 台東縣 021 花蓮縣 022 澎湖縣 023 金門縣 024 連江縣 025 南海諸島 026 釣魚台列嶼 027 屏東縣琉球鄉離島 028 台東縣綠島、蘭嶼 029
	ATMInfo	ATM 資訊	JSON	ChoosePaymentList 如選擇 0 或 3，則必填
	*ExpireDate	允許繳費有效 天數	Int	若需設定最長 60 天，最短 1 天。未設定此參數則預設為 3 天 注意事項： 以天為單位
	ATMBankCode	ATM 銀行代碼	String(10)	繳費銀行代碼 若未傳入，依系統預設銀行為主。

				請參考 銀行代碼表
CVSInfo	超商代碼資訊	JSON	ChoosePaymentList	如選擇 0 或 4，則必填
*StoreExpireDate	超商繳費截止時間	Int		以分鐘為單位 若未設定此參數，預設為 10080 分鐘(7 天)。若需設定此參數，請於建立訂單時將此參數送給綠界。 提醒您，CVS 帶入數值不可超過 86400 分鐘，超過時一律以 86400 分鐘計(60 天)例：08/01 的 20:15 分購買商品，繳費期限為 7 天，表示 08/08 的 20:15 分前您必須前往超商繳費。
CVSCode	超商代碼	String(10)		CVS: 超商代碼繳款(不指定超商) OK: OK 超商代碼繳款 FAMILY: 全家超商代碼繳款 HILIFE: 萊爾富超商代碼繳款 IBON: 7-11 ibon 代碼繳款 若未傳入，預設為 CVS
Desc_1	交易描述 1	String(20)		若繳費超商為 family(全家)或 ibon(7-11)時，會顯示在超商繳費平台螢幕上
Desc_2	交易描述 2	String(20)		若繳費超商為 family(全家)或 ibon(7-11)時，會顯示在超商繳費平台螢幕上
Desc_3	交易描述 3	String(20)		若繳費超商為 family(全家)或 ibon(7-11)時，會顯示在超商繳費平台螢幕上
Desc_4	交易描述 4	String(20)		若繳費超商為 family(全家)或 ibon(7-11)時，會顯示在超商繳費平台螢幕上
BarcodeInfo	超商條碼資訊	JSON	ChoosePaymentList	如選擇 0 或 5，則必填
*StoreExpireDate	超商繳費截止時間	Int		以天為單位，預設為 7 天。
ConsumerInfo	消費者資訊	JSON		
MerchantMemberID	消費者會員編號	String(60)		當[RememberCard] = 1，此欄位必填
Email	信用卡持卡人電子信箱	String(30)		
Phone	信用卡持卡人電話	String(60)		注意事項： 可接受帶國碼的形式，但不可帶+號 ex: 886912345678
Name	信用卡持卡人姓名	String(50)		注意事項： 該欄位可接受中文、英文與部分特殊符號。 可支援的特殊符號為: , . () / -
CountryCode	國別碼	String(3)		持卡人帳單地址國別碼，請參考 ISO3166 。臺灣請填寫 158

Address	地址	String(50)	持卡人帳單地址
UnionPayInfo	銀聯卡資訊	JSON	以下情況為必填: ChoosePaymentList 為 0 或 6 時
OrderResultURL	銀聯卡驗證回傳 付款結果網址	String(200)	使用 APP SDK 時，銀聯卡驗證結果將於 SDK callback 回傳。 此欄位於 APP SDK 目前無作用，請固定填入 https://www.ecpay.com.tw/ 即可
CustomField	自訂欄位	String(200)	提供特店使用客制化欄位

Data 範例：

```
{
  "MerchantID": "3002607",
  "RememberCard": 1,
  "PaymentUIType": 2,
  "ChoosePaymentList": "1,2,3",
  "OrderInfo": {
    "MerchantTradeNo": "20180914001",
    "MerchantTradeDate": "2020/09/26 14:49:12",
    "TotalAmount": 100,
    "ReturnURL": "https://yourReturnURL.com"
  },
  "CardInfo": {
    "OrderResultURL": "https://yourOrderResultURL.com",
    "CreditInstallment": "3,6,9,12"
  },
  "ATMInfo": {
    "ExpireDate": 3
  },
  "ConsumerInfo": {
    "MerchantMemberID": "test123456",
    "Email": "customer@email.com",
    "Phone": "0912345678",
    "Name": "Test",
    "CountryCode": "158"
  }
}
```

- 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
RqHeader	傳輸資料	JSON	
	Timestamp	Number	時間戳 Unix timestamp 注意事項: 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RqHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
  "Data": "..."
}
```

- Data 參數說明：請在解密後對參數值進行 URLDecode

參數	參數名稱	型態	說明
RtnCode	交易狀態	Int	1 代表 API 執行成功，其餘代碼均為失敗。 失敗代碼請參考 交易訊息代碼表
RtnMsg	回應訊息	String(200)	
RtnCode 為 1 時，以下參數才有值:			
PlatformID	平台商編號	String(10)	特約合作平台商特店編號。
MerchantID	特店編號	String(10)	
Token	交易代碼	String(64)	可使用的效期為 30 分鐘
TokenExpireDate	交易代碼到期時間	String(20)	yyyy/MM/dd HH:mm:ss

Data 解密後:

```
{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "PlatformID": "1234567890",
  "MerchantID": "1234567890",
  "Token": "m12dae4846446sq",
  "TokenExpireDate": "2020/09/18 15:39:10"
}
```

● 刪除信用卡

- 應用場景：使用者可於交易中進行記憶卡號功能，當使用者想要刪除信用卡資料，廠商可透過此功能刪除已綁定的信用卡。
- 介接路徑
 - 正式環境：https://ecpg.ecpay.com.tw/Merchant/GetTokenbyUser
 - 測試環境：https://ecpg-stage.ecpay.com.tw/Merchant/GetTokenbyUser
- 廠商傳入參數說明：(JSON 格式，名稱若有紅色星號*為必填欄位)
 - Content Type : application/json
 - HTTP Method : POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
*Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
*Revision	串接文件版號	String(10)	請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RqHeader": {
    "Timestamp": 1525168923,
    "Revision": "1.0.0"
  },
  "Data": "..."
}
```

- Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	特約合作平台商代號	String(10)	
*MerchantID	特店編號	String(10)	
*ConsumerInfo	消費者資訊	JSON	
*MerchantMemberID	消費者會員編號	String(60)	
Email	信用卡持卡人電子信箱	String(30)	
Phone	信用卡持卡人電話	String(60)	
Name	信用卡持卡人姓名	String(50)	
CountryCode	國別碼	String(3)	持卡人帳單地址國別碼 · IOS3166
Address	地址	String(50)	持卡人帳單地址
CustomField	自訂欄位	String(200)	提供特店使用客制化欄位

Data 範例：

```
{
  "MerchantID": "3002607",
  "ConsumerInfo": {
    "MerchantMemberID": "test123466",
    "Email": "customer@email.com",
    "Phone": "0912345678",
    "Name": "Test",
    "CountryCode": "158"
  },
  "ReturnURL": "https://aa.aa.aa"
}
```


● 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
RqHeader	傳輸資料	JSON	
	Timestamp	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RqHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例：

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
  "Data": "..."
}
```

● Data 參數說明：請在解密後對參數值進行 URLDecode

參數	參數名稱	型態	說明
RtnCode	交易狀態	Int	1 代表 API 執行成功，其餘代碼均為失敗。 失敗代碼請參考 交易訊息代碼表
RtnMsg	回應訊息	String(200)	
RtnCode 為 1 時，以下參數才有值：			
PlatformID	平台商編號	String(10)	特約合作平台商特店編號。
MerchantID	特店編號	String(10)	
Token	交易代碼	String(64)	可使用的效期為 30 分鐘
TokenExpireDate	交易代碼到期時間	String(20)	yyyy/MM/dd HH:mm:ss

Data 解密後:

```
{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "PlatformID": "1234567890",
  "MerchantID": "1234567890",
  "Token": "m12dae4846446sq",
  "TokenExpireDate": "2020/09/18 15:39:10"
}
```

6. 站內付 2.0 Android SDK(App)

- 應用場景:

適用於前端 Android App，提供廠商站內付 2.0 金流功能。Android SDK 詳細的安裝方式與範例程式請參閱 [GitHub](#)

- 使用方式:

1. **Import:**

```
import tw.com.ecpay.paymentgatewaykit.manager.*
```

2. **Initialize:** 初始化 SDK，帶入 Activity 與環境參數。

```
PaymentkitManager.initialize(mActivity, serverType);
```

■ Input:

參數	參數名稱	型態	說明
*Activity	Activity 元件	Android Activity	請提供 APP 當前畫面的 Activity 元件
*ServerType	環境參數	String	Stage:測試環境 Prod:正式環境

3. **CreatePayment:** 產生站內付 2.0 付款功能

```
PaymentkitManager.createPayment(mActivity, mFragment,
    mModel.token.get(), languageCode, useResultPage,
    mExampleData.getAppName(), PaymentkitManager.RequestCode_CreatePayment);
```

■ Input:

參數	參數名稱	型態	說明
*Activity	Activity 元件	Android Activity	請提供 APP 當前畫面的 Activity 元件
Fragment	Fragment 元件	Android Fragment	如果 APP 當前畫面的切換方式為 Fragment Fragment 為必填 如未填此欄位，該 Fragment 會收不到付款結果資料
*Token	廠商驗證碼	String	初始化站內付 2.0 畫面代碼 此參數具 30 分鐘時效性

*Language	多語系	String	可使用以下方式設定使用語系 繁中: zh-TW 英語: en-US
*UseResultPage	是否顯示付款結果	Int	是否由綠界顯示付款結果通知或超商/ATM 取號結果通知頁面 0 : 否 1 : 是
*AppStoreName	APP 商店名稱	String	APP 商店名稱,顯示於付款畫面的 APP 商店名稱,特店可自行定義顯示的名稱
*RequestCode	畫面返回值	Int	當信用卡刷卡畫面關閉時,返回當前 Activity/ Fragment 的值

4. CreatePaymentResult: 需與 CreatePayment 一起使用

```

public void onActivityResult(int requestCode, int resultCode, Intent data) {
    Utility.log("onActivityResult(), requestCode:" + requestCode + ", resultCode:" + resultCode);
    if(requestCode == PaymentkitManager.RequestCode_CreatePayment) {
        PaymentkitManager.createPaymentResult(mActivity, resultCode, data, new
        CallbackFunction<CreatePaymentCallbackData>() {
            @Override
            public void callback(CreatePaymentCallbackData callbackData) {
                switch (callbackData.getCallbackStatus()) {
                    case Success:
                        //enter your code
                    case Fail:
                        // enter your code
                }
            }
        });
    }
}

```

■ Input:

參數	參數名稱	型態	說明
*Activity	Activity 元件	Android Activity	請提供 APP 當前畫面的 Activity 元件
*ResultCode	Activity result code	Int	Activity result code
*Data	Android Intent	Intent	Android Intent

*Callback	Callback	CallbackFunction	此非同步工作執行完成後的後續邏輯處理
-----------	----------	------------------	--------------------

■ Callback:

參數	參數名稱	型態	說明
CallbackStatus	Callback 狀態	enum CallbackStatus	※注意事項: 當狀態為 Cancel 時，建議利用 查詢訂單 API 反查訂單狀態。
當 CallbackStatus 為 Success 時，將會回傳以下參數:			
RtnCode	交易狀態	Int	1 代表 API 執行成功，其餘代碼均為失敗。 失敗代碼請參考 交易訊息代碼表
RtnMsg	回應訊息	String(200)	
PaymentType	付款方式	enum PaymentType	
PlatformID	平台商編號	String(10)	特約合作平台商特店編號。
MerchantID	特店編號	String(10)	
OrderInfo	訂單資訊	JSON Object	
MerchantTradeNo	特店交易編號	String(20)	
TradeNo	綠界交易編號	String(20)	
TradeAmt	交易金額	Int	
TradeDate	訂單成立時間	String(20)	yyyy/MM/dd HH:mm:ss
PaymentType	付款方式	String(20)	
PaymentDate	付款時間	String(20)	yyyy/MM/dd HH:mm:ss
ChargeFee	手續費	Number	
TradeStatus	交易狀態	String(8)	
CardInfo	信用卡授權資訊	JSON Object	
AuthCode	銀行授權碼	String(6)	
Gwsr	授權交易單號	Int	
ProcessDate	交易時間	String(20)	yyyy/MM/dd HH:mm:ss
Amount	金額	Int	
Stage	分期期數	Int	
Stast	首期金額	Int	
Staed	各期金額	Int	
Eci	3D(VBV) 回傳值	Int	Eci=5,6,2,1 代表該筆交易為 3D 交易
Card6No	信用卡卡號前六碼	String(6)	
Card4No	信用卡卡號末四碼	String(4)	

RedDan	紅利扣點	Int	使用信用卡紅利時回傳
RedDeAmt	紅利折抵金額	Int	使用信用卡紅利時回傳
RedOkAmt	實際扣款金額	Int	使用信用卡紅利時回傳
RedYet	紅利剩餘點數	Int	使用信用卡紅利時回傳
PeriodType	週期種類	String (1)	定期定額時回傳，訂單建立時所設定的週期種類
Frequency	執行頻率	Int	定期定額時回傳，訂單建立時所設定的執行頻率
ExecTimes	執行次數	Int	定期定額時回傳，訂單建立時所設定的執行頻率
PeriodAmount	訂單建立時的每次要授權金額	Int	定期定額時回傳
TotalSuccessTimes	目前已成功授權的次數	Int	定期定額時回傳，目前已成功授權的次數。
TotalSuccessAmount	目前已成功授權的金額合計	Int	定期定額時回傳
ATMInfo		ATM 資訊	JSON Object
BankCode	繳費銀行代碼	String (3)	
vAccount	繳費虛擬帳號	String (16)	
ExpireDate	繳費期限	String (10)	格式為 yyyy/MM/dd
CVSInfo		超商代碼資訊	JSON Object
PaymentNo	繳費代碼	String(14)	
ExpireDate	繳費期限	String(20)	格式為 yyyy/MM/dd HH:mm:ss
PaymentURL	繳費連結	String(100)	由綠界科技提供手機上顯示的三段式繳費條碼網頁
BarcodeInfo		超商條碼資訊	JSON Object
ExpireDate	繳費期限	String(20)	格式為 yyyy/MM/dd HH:mm:ss
Barcode1	條碼第一段號碼	String(20)	格式為 9 碼數字
Barcode2	條碼第二段號碼	String(20)	格式為 16 碼數字
Barcode3	條碼第三段號碼	String(20)	格式為 15 碼數字
CoBrandingInfo		聯名卡資料	Array
CoBrandingCode	聯名卡代號	String(20)	廠商後台設定的聯名卡代號
Comment	備註	String(100)	
CustomField	自訂欄位	String(200)	提供特店使用客制化欄位

7. 站內付 2.0 iOS SDK(App)

- 應用場景:

適用於前端 iOS App，提供廠商站內付 2.0 金流功能。iOS SDK 詳細的安裝方式與範例程式請參閱 [GitHub](#)

- 支援:

1. XCode 12.2+
2. Swift 5+
3. iOS 10+
4. Cocoapods 1.10.0+

- 使用方式:

1. **Import:**

```
import ECPayPaymentGatewayKit
```

2. **Initialize:** 建議於 AppDelegate.swift 內執行初始化的動作，環境參數請參考下段說明。

```
ECPayPaymentGatewayManager.sharedInstance().initialize(env: .Stage)
```

- Input:

參數	參數名稱	型態	說明
*Env	環境參數	String	Stage:測試環境, Prod:正式環境

3. CreatePayment: 產生站內付 2.0 功能畫面

```
ECPayPaymentManager.sharedInstance().createPayment(  
    token: token,  
    language: "zh-TW",  
    useResultPage: 1,  
    appStoreName: "測試商店"  
) { (state) in  
    if let creditPayment:CreditPaymentCallbackState = state as? CreditPaymentCallbackState {  
        if let order = creditPayment.OrderInfo {  
            print(order)  
        }  
        if let card = creditPayment.CardInfo {  
            print(card)  
        }  
        if let aTM = creditPayment.ATMInfo {  
            print(aTM)  
        }  
        if let cVS = creditPayment.CVSInfo {  
            print(cVS)  
        }  
        if let barcode = creditPayment.BarcodeInfo {  
            print(barcode)  
        }  
    }  
    switch state.callbackStateStatus {  
        case .Success:  
            print("Success")  
        case .Fail:  
            print("Fail")  
        case .Cancel:  
            print("Cancel")  
        case .Unknown:  
            print("Unknown")  
    }  
}
```


■ Input:

參數	參數名稱	型態	說明
*Token	廠商驗證碼	String	初始化站內付 2.0 畫面代碼 此參數具 30 分鐘時效性
*Language	多語系	String	可使用以下方式設定使用語系 繁中: zh-TW 英語: en-US
*UseResultPage	是否顯示付款結果	Int	是否由綠界顯示付款結果通知或超商/ATM 取號結果通知頁面 0 : 否 1 : 是
*AppStoreName	APP 商店名稱	String	APP 商店名稱,顯示於付款畫面的 APP 商店名稱,特店可自行定義顯示的名稱

■ Callback:

參數	參數名稱	型態	說明
callbackStateStatus	Callback 狀態	enum CallbackStateStatusType	※注意事項: 當狀態為 Cancel 時,建議利用 查詢訂單 API 反查訂單狀態。
callbackStateMessage	回應訊息	String	回應訊息
當 CallbackStatus 為 Success 時,將會回傳以下參數:			
RtnCode	交易狀態	Int	1 代表 API 執行成功,其餘代碼均為失敗。 失敗代碼請參考 交易訊息代碼表
RtnMsg	回應訊息	String(200)	
PaymentType	付款方式	enum PaymentType	
PlatformID	平台商編號	String(10)	特約合作平台商特店編號。
MerchantID	特店編號	String(10)	
OrderInfo	訂單資訊	JSON Object	
MerchantTradeNo	特店交易編號	String(20)	
TradeNo	綠界交易編號	String(20)	
TradeAmt	交易金額	Int	
TradeDate	訂單成立時間	String(20)	yyyy/MM/dd HH:mm:ss
PaymentType	付款方式	String(20)	
PaymentDate	付款時間	String(20)	yyyy/MM/dd HH:mm:ss

ChargeFee	手續費	Number	
TradeStatus	交易狀態	String(8)	
CardInfo	信用卡授權資訊	JSON Object	
AuthCode	銀行授權碼	String(6)	
Gwsr	授權交易單號	Int	
ProcessDate	交易時間	String(20)	yyyy/MM/dd HH:mm:ss
Amount	金額	Int	
Stage	分期期數	Int	
Stast	首期金額	Int	
Staed	各期金額	Int	
Eci	3D(VBV) 回傳值	Int	Eci=5,6,2,1 代表該筆交易為 3D 交易
Card6No	信用卡卡號前六碼	String(6)	
Card4No	信用卡卡號末四碼	String(4)	
RedDan	紅利扣點	Int	使用信用卡紅利時回傳
RedDeAmt	紅利折抵金額	Int	使用信用卡紅利時回傳
RedOkAmt	實際扣款金額	Int	使用信用卡紅利時回傳
RedYet	紅利剩餘點數	Int	使用信用卡紅利時回傳
PeriodType	週期種類	String (1)	定期定額時回傳·訂單建立時所設定的週期種類
Frequency	執行頻率	Int	定期定額時回傳·訂單建立時所設定的執行頻率
ExecTimes	執行次數	Int	定期定額時回傳·訂單建立時所設定的執行頻率
PeriodAmount	訂單建立時的每次要授權金額	Int	定期定額時回傳
TotalSuccessTimes	目前已成功授權的次數	Int	定期定額時回傳·目前已成功授權的次數。
TotalSuccessAmount	目前已成功授權的金額合計	Int	定期定額時回傳
ATMInfo	ATM 資訊	JSON Object	
BankCode	繳費銀行代碼	String (3)	
vAccount	繳費虛擬帳號	String (16)	
ExpireDate	繳費期限	String (10)	格式為 yyyy/MM/dd

CVSInfo		超商代碼資訊	JSON Object	
	PaymentNo	繳費代碼	String(14)	
	ExpireDate	繳費期限	String(20)	格式為 yyyy/MM/dd HH:mm:ss
	PaymentURL	繳費連結	String(100)	由綠界科技提供手機上顯示的三段式繳費條碼網頁
BarcodeInfo		超商條碼資訊	JSON Object	
	ExpireDate	繳費期限	String(20)	格式為 yyyy/MM/dd HH:mm:ss
	Barcode1	條碼第一段號碼	String(20)	格式為 9 碼數字
	Barcode2	條碼第二段號碼	String(20)	格式為 16 碼數字
	Barcode3	條碼第三段號碼	String(20)	格式為 15 碼數字
CoBrandingInfo		聯名卡資料	Array	如果此筆訂單付款的信用卡為廠商設定之聯名卡，將會回傳此欄位資料
	CoBrandingCode	聯名卡代號	String(20)	廠商後台設定的聯名卡代號
	Comment	備註	String(100)	
CustomField		自訂欄位	String(200)	提供特店使用客制化欄位

8. 付款結果通知

- 應用場景 I：

- Server 端方式(Server POST)([ReturnURL](#))

當消費者付款完成後，廠商接受綠界的付款結果訊息，並回應接收訊息

Step1.綠界：以 ServerPost 方式傳送付款結果訊息至廠商的 Server 網址([ReturnURL](#))

Step2.廠商：收到綠界的付款結果訊息，回應 1|OK

注意事項:

1.超商繳費條碼(BARCODE)因銀行端作業時間關係會於消費者付款完成兩天後回傳。

2.ATM、CVS、BARCODE 可透過廠商管理後台的『模擬付款』，來確認 ReturnURL 是否正確接收付款結果通知。

3.付款結果通知常見付款狀態未更新原因說明

4.1|OK 僅是廠商回應綠界是否收到通知，並不會改變付款狀態。

- 綠界回傳參數說明 I ([ReturnURL](#))：

- Accept : text/html
 - Content Type : application/ json
 - HTTP Method : POST

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號
RpHeader	回傳資料	JSON	
Timestamp	回傳時間	Number	時間戳 Unix timestamp
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RpHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
  "Data": "..."
```

}

● Data 參數說明：

參數	參數名稱	型態	說明
RtnCode	交易狀態	Int	1 代表 API 執行成功，其餘代碼均為失敗。 失敗代碼請參考 交易訊息代碼表
RtnMsg	回應訊息	String(200)	
RtnCode 為 1 時，以下參數才会有值：			
PlatformID	平台商編號	String(10)	特約合作平台商特店編號。
MerchantID	特店編號	String(10)	
SimulatePaid	是否為模擬付款	Int	當交易為模擬付款時，才會回傳此欄位 回傳值： 1:代表此交易為模擬付款，RtnCode 也為 1。並非是由消費者實際真的付款，所以綠界也不會撥款給廠商，請勿對該筆交易做出貨等動作，以避免損失。 注意事項： 1.特店可透過廠商後台來針對單筆訂單模擬綠界回傳付款通知，以方便介接 API。 2.此功能僅只是用於測試 ReturnURL 是否能成功接收，不會改變付款狀態。 3.只有透過廠商後台的定期定額查詢功能發動的模擬付款通知，綠界才會傳送此參數，正常由定期定額排程所發送的付款通知，不會傳送此參數。
OrderInfo			
	MerchantTradeNo	特店交易編號	String(20)
	TradeNo	綠界的交易編號	String(20) 請保存綠界的交易編號與特店交易編號[MerchantTradeNo]的關聯。
	PaymentDate	付款時間	String(20) 格式為 yyyy/MM/dd HH:mm:ss
	TradeAmt	交易金額	Int
	PaymentType	特店選擇的付款方式	String(20)
	TradeDate	訂單成立時間	String(20) yyyy/MM/dd HH:mm:ss
	ChargeFee	手續費	Number
	TradeStatus	交易狀態	String(8) 回傳值： 若為 0 時，代表交易訂單成立未付款 若為 1 時，代表交易訂單成立已付款

CVSInfo		CVS 資訊	JSON Object	
	PayFrom	繳費超商	String(10)	family：全家 hilife：萊爾富 okmart：OK 超商 ibon：7-11
	PaymentNo	繳費代碼	String(14)	
	PaymentURL	繳費連結	String(100)	
BarcodeInfo		Barcode 資訊	JSON Object	
	PayFrom	繳費超商	String(10)	family：全家 hilife：萊爾富 okmart：OK 超商 ibon：7-11
ATMInfo		ATM 資訊	JSON Object	
	ATMAccBank	付款人銀行代碼	String(3)	
	ATMAccNo	付款人銀行帳號 後五碼	String(5)	
CardInfo		信用卡授權資訊		
	AuthCode	銀行授權碼	String(6)	
	Gwsr	授權交易單號	Int	
	ProcessDate	交易時間	String(20)	yyyy/MM/dd HH:mm:ss
	Amount	金額	Int	
	Eci	3D(VBV) 回傳值	Int	Eci=5,6,2,1 代表該筆交易為 3D 交易
	Card6No	信用卡卡號前六碼	String(6)	
	Card4No	信用卡卡號末四碼	String(4)	
	Stage	分期期數	Int	分期付款時回傳
	Stast	首期金額	Int	分期付款時回傳
	Staed	各期金額	Int	分期付款時回傳
	RedDan	紅利扣點	Int	紅利折抵時回傳
	RedDeAmt	紅利折抵金額	Int	紅利折抵時回傳
	RedOkAmt	實際扣款金額	Int	紅利折抵時回傳
	RedYet	紅利剩餘點數	Int	紅利折抵時回傳
	PeriodType	訂單建立時的所 設定的週期種類	String(1)	定期定額時才會回傳
	Frequency	訂單建立時的所	Int	定期定額時才會回傳

		設定的執行頻率		
	ExecTimes	訂單建立時的所設定的執行次數	Int	定期定額時才會回傳
	PeriodAmount	訂單建立時的每次要授權金額	Int	定期定額時才會回傳
	TotalSuccessTimes	目前已成功授權的次數	Int	定期定額時才會回傳
	TotalSuccessAmount	目前已成功授權的金額合計	Int	定期定額時才會回傳
CoBrandingInfo		聯名卡資料	Array	如果此筆訂單付款的信用卡為廠商設定之聯名卡，將會回傳此欄位資料
	CoBrandingCode	聯名卡代號	String(20)	廠商後台設定的聯名卡代號
	Comment	備註	String(100)	
CustomField		自訂欄位	String(200)	提供特店使用客制化欄位

Data 解密後:

```

{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "PlatformID": "1234567890",
  "MerchantID": "1234567890",
  "SimulatePaid": 1,
  "OrderInfo": {
    "MerchantTradeNo": "test123466",
    "TradeNo": "123456789",
    "TradeDate": "2020/09/18 15:39:10"
  },
  "ThreeDInfo": {
    "ThreeDURL": "https://3durl.com.tw"
  }
}

```

● 應用場景 II：

當訂單是使用信用卡定期定額的交易時，在每次授權成功後，綠界會傳送付款通知結果

Step 1. 綠界：第二次授權是由綠界的排程進行授權，交易授權成功後以 Server POST 方式傳送付款通知

至廠商傳送的付款完成通知回傳網址[[PeriodReturnURL](#)]。

Step 2. 廠商：收到綠界的付款通知訊息後，需比對檢查碼是否相符。

注意事項：

- (1) 第一次授權成功時，付款結果參數會回傳到付款完成通知回傳網址[ReturnURL]，
- (2) 第二次授權成功是由綠界的排程進行授權，所以從第二次開始的資料，都是回傳到所設定的付款完成通知回傳網址[PeriodReturnURL]。

● 綠界回傳參數說明 II：

- Accept : text/html
- Content Type : application/json
- HTTP Method : POST

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號
RpHeader	回傳資料	JSON	
Timestamp	回傳時間	Number	時間戳 Unix timestamp
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RqHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
  "Data": "..."
}
```

● Data 參數說明：

參數	參數名稱	型態	說明
RtnCode	交易狀態	Int	1 代表 API 執行成功，其餘代碼均為失敗。

			失敗代碼請參考 交易訊息代碼表
RtnMsg	回應訊息	String(200)	
以下為 RtnCode 為 1 時才會回傳			
PlatformID	平台商編號	String(10)	特約合作平台商特店編號。
MerchantID	特店編號	String(10)	
OrderInfo	訂單資訊	JSON Object	
MerchantTradeNo	特店交易編號	String(20)	
TradeNo	綠界的交易編號	String(20)	請保存綠界的交易編號與特店交易編號[MerchantTradeNo]的關聯。
PaymentDate	付款時間	String(20)	格式為 yyyy/MM/dd HH:mm:ss
TradeAmt	交易金額	Int	
PaymentType	特店選擇的付款方式	String(20)	
TradeDate	訂單成立時間	String(20)	yyyy/MM/dd HH:mm:ss
ChargeFee	手續費	Number	
TradeStatus	交易狀態	String(8)	回傳值： 若為 0 時，代表交易訂單成立未付款 若為 1 時，代表交易訂單成立已付款 若為 10200095 時，代表消費者未選擇付款方式，故交易失敗。
CardInfo	授權資訊	JSON Object	
AuthCode	銀行授權碼	String(6)	
Gwsr	授權交易單號	Int	
ProcessDate	交易時間	String(20)	yyyy/MM/dd HH:mm:ss
Amount	金額	Int	
Eci	3D(VBV) 回傳值	Int	Eci=5,6,2,1 代表該筆交易為 3D 交易
Card6No	信用卡卡號前六碼	String(6)	
Card4No	信用卡卡號末四碼	String(4)	
PeriodType	訂單建立時的所設定的週期種類	String(1)	
Frequency	訂單建立時的所設定的執行頻率	Int	
ExecTimes	訂單建立時的所設定的執行次數	Int	

PeriodAmount	訂單建立時的每次要授權金額	Int	
TotalSuccessTimes	目前已成功授權的次數	Int	
TotalSuccessAmount	目前已成功授權的金額合計	Int	
CustomField	自訂欄位	String(200)	提供特店使用客制化欄位

Data 解密後:

```
{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "PlatformID": "1234567890",
  "MerchantID": "1234567890",
  "OrderInfo": {
    "MerchantTradeNo": "test123466",
    "TradeNo": "123456789",
    "TradeDate": "2020/09/18 15:39:10"
  },
  "ThreeDInfo": {
    "ThreeDURL": "https://3durl.com.tw"
  }
}
```

注意事項：

- (1) [\[PeriodReturnURL\]](#)每期僅會收到一次通知，如未收到通知，請利用[信用卡定期定額訂單查詢](#)取得實際授權結果。
- (2) 若遇消費者已付款，但未收到綠界回傳付款完成資訊，此時請檢查接收回傳參數的伺服器是否服務正常，導致無法接收。請參考：[無法收到綠界回傳的付款結果通知](#)
- (3) 廠商務必判斷交易狀態[RtnCode]是否為 1，若非 1 時請勿對該筆交易做出貨動作，並取得交易訊息 [RtnMsg] 記錄失敗原因。

9. 查詢訂單(Server)

- 應用場景:

提供廠商查詢訂單資訊，可透過此 API 來過濾是否為有效訂單。

- 介接路徑：

- 正式環境：https://ecpayment.ecpay.com.tw/1.0.0/Cashier/QueryTrade
- 測試環境：https://ecpayment-**stage**.ecpay.com.tw/1.0.0/Cashier/QueryTrade

- 廠商傳入參數說明：(JSON 格式)

- Content Type : application/json
- HTTP Method : POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
*Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
*Revision	串接版號	String(10)	請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料		加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RqHeader": {
    "Timestamp": 1525168923,
    "Revision": "1.0.0"
  },
  "Data": "..."
}
```

- Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	平台商代號	String(10)	平台商特店編號
*MerchantID	特店編號	String(10)	
*MerchantTradeNo	特店交易編號	String(20)	

Data 範例：

```
{
  "MerchantID": "3002607",
  "MerchantTradeNo": "20180914001"
}
```

- 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	
RpHeader	回傳資料	JSON	
Timestamp	回傳時間	Number	時間戳 Unix timestamp
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RpHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料		加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "成功",
  "Data": "..."
}
```

● Data 參數說明：請在解密後對參數值進行 URLDecode

參數	參數名稱	型態	說明
RtnCode	交易狀態	Int	1 代表 API 執行成功，其餘代碼均為失敗。 失敗代碼請參考交易訊息代碼表
RtnMsg	回應訊息	String(200)	
RtnCode 為 1 時，以下參數才有值：			
PlatformID	平台商代號	String(10)	平台商特店編號
MerchantID	特店編號	String(10)	
OrderInfo	訂單資訊	JSON	
MerchantTradeNo	特店交易編號	String(20)	不可重複
TradeNo	綠界交易編號	String(20)	
TradeAmt	交易金額	Int	
TradeDate	訂單成立時間	String(20)	yyyy/MM/dd HH:mm:ss
PaymentType	付款方式	String(20)	
PaymentDate	付款時間	String(20)	yyyy/MM/dd HH:mm:ss
ChargeFee	手續費	Number	格式為整數 8 位，小數 2 位
TradeStatus	交易狀態	String(8)	回傳值： 若為 0 時，代表交易訂單付款失敗 若為 1 時，代表交易訂單付款成功
CardInfo	授權資訊	JSON	
AuthCode	銀行授權碼	String(6)	
Gwsr	授權交易單號	Int	
ProcessDate	交易時間	String(20)	yyyy/MM/dd HH:mm:ss
Amount	金額	Int	
Stage	分期期數	Int	
Stast	首期金額	Int	
Staed	各期金額	Int	
Eci	3D(VBV) 回傳值	Int	Eci=5,6,2,1 代表該筆交易為 3D 交易
Card6No	信用卡卡號前六碼	String(6)	
Card4No	信用卡卡號末四碼	String(4)	
RedDan	紅利扣點	Int	紅利折抵時回傳
RedOkAmt	實際扣款金額	Int	紅利折抵時回傳
RedYet	紅利剩餘點數	Int	紅利折抵時回傳

PeriodType	週期種類	String (1)	定期定額時回傳，訂單建立時所設定的週期種類 D：以天為週期，M：以月為週期，Y：以年為週期
Frequency	執行頻率	Int	定期定額時回傳，訂單建立時所設定的執行頻率
ExecTimes	執行次數	Int	定期定額時回傳，訂單建立時所設定的執行次數
PeriodAmount	訂單建立時的每次要授權金額	Int	定期定額時回傳
TotalSuccessTimes	目前已成功授權的次數	Int	定期定額時回傳，目前已成功授權的次數。
TotalSuccessAmount	目前已成功授權的金額合計	Int	定期定額時回傳
CVSInfo	CVS 資訊	JSON	
PayFrom	繳費超商	String(10)	
PaymentNo	繳費代碼	String(14)	
PaymentURL	繳費連結	String(14)	
BarcodeInfo	Barcode 資訊	JSON	
PayFrom	繳費超商	String(10)	
ATMInfo	ATM 資訊	JSON	
ATMAccBank	付款人銀行代碼	String(3)	
ATMAccNo	付款人銀行帳號後五碼	String(5)	
CoBrandingInfo	聯名卡資料	Array	如果此筆訂單付款的信用卡為特店設定之聯名卡，將會回傳此欄位資料
CoBrandingCode	聯名卡代號	String(20)	綠界廠商後台設定的聯名卡代號
Comment	備註	String(100)	
CustomField	自訂欄位	String(200)	提供特店使用客制化欄位

Data 範例：

```
{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "MerchantID": "3002607",
  "OrderInfo": {
    "MerchantTradeNo": "20180914001",
    "TradeNo": "1809261503338172",
    "TradeDate": "2018/09/26 14:59:54"
  }
}
```

```
"CardInfo": {  
  "Gwsr": 10735183,  
  "ProcessDate": "2018/09/26 14:59:54",  
  "AuthCode": "777777",  
  "Amount": 100,  
  "Eci": 2,  
  "Card4No": "2222",  
  "Card6No": "491122",  
  "RedDan": 0,  
  "RedOkAmt": 0,  
  "RedYet": 0  
}  
}
```

10. 查詢信用卡單筆明細記錄(Server)

- 應用場景：

提供查詢信用卡明細記錄。

- 介接路徑：

- 正式環境：<https://ecpayment.ecpay.com.tw/1.0.0/CreditDetail/QueryTrade>

- 測試環境：<https://ecpayment-stage.ecpay.com.tw/1.0.0/CreditDetail/QueryTrade>

- 廠商傳入參數說明：(JSON 格式)

- Content Type : application/json

- HTTP Method : POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
*Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
*Revision	串接版號	String(10)	請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RqHeader": {
    "Timestamp": 1525168923,
    "Revision": "1.0.0"
  },
  "Data": "..."
}
```


- Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	平台商編號	String(10)	特約合作平台商特店編號
*MerchantID	特店編號	String(10)	
*MerchantTradeNo	特店交易編號	String(20)	
TradeNo	綠界交易編號	String(20)	此欄位可從 定期定額訂單查詢 API 取得。可查詢定期定額第一筆之後的授權資料內容。

Data 範例：

```
{
  "MerchantID": "3002607",
  "PlatformID": "2000132",
  "MerchantTradeNo": "123486456"
}
```

- 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號
RpHeader	傳輸資料	JSON	
Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項: 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RpHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
}
```

```
"Data": "..."  
}
```

Data 參數說明：請在解密後對參數值進行 URLDecode

參數	參數名稱	型態	說明
RtnMsg	回應訊息	String	若成功回傳空值其餘則為失敗 若查無商家或商家已到期，回傳值為： error_Stop 若查無該筆交易授權單號時，回傳值為： error_nopay 若發生錯誤或資料檢核失敗，則回傳值為： error 若發生傳遞的資料長度不對，則回傳值為： error_Length 若發生金額參數不對，則回傳值為： error_Amount 若發生該筆交易帳目運作中，無法由程式關帳，則回傳值為： error_close
RtnValue	回應資料	JSON	RtnMsg 為空值時才會回傳
RtnValue 回應內容			
TradeID	授權單號	Int	
Amount	交易金額	Int	
ClsAmt	已關帳金額	Int	
AuthTime	訂單成立時間	String(24)	
Status	交易狀態	String(30)	若未有關帳明細交易，狀態回傳值如下： 已取消(Canceled)：此筆交易已取消 未授權(Unauthorized)：銀行未授權完成 已授權(Authorized)：銀行已完成授權 若已有關帳明細交易，狀態回傳值如下： 銀行拒絕(Bank Refused) 要關帳(To be captured) 關帳中(Captured in progress) 已關帳(Captured) 要取消(To be canceled) 取消中(Canceled in progress) 銀行追回中(Bank chargebacked in progress) 銀行已追回(Bank chargebacked) 不明(Unknown) 操作取消(Operation canceled)
CloseData	交易明細	Array	
	Status	狀態	狀態回傳值如下： 銀行拒絕(Bank Refused)

				取消中(Canceled in progress) 銀行追回中(Bank chargebacked in progress) 銀行已追回(Bank chargebacked) 批次失敗(Batch failed) 不明(Unknown) 操作取消(Operation canceled)
	Amount	交易金額	Int	
	DateTime	關帳時間	String(24)	

Data 解密後：

```

{
  "RtnMsg": "",
  "RtnValue": {
    "TradeID ": 245678,
    "Amount ":100,
    "ClsAmt ": 100,
    "AuthTime": "2018/09/25 14:59:54",
    "Status": "To be captured"
  },
  "CloseData": [{
    "Status": "Bank chargebacked ",
    "Amount": 100,
    "DateTime" : "2018/09/26 14:59:54"
  }]
}

```

11. 信用卡請退款功能(Server)

- 應用場景：

當收到訂單授權成功，透由 [查詢信用卡單筆明細記錄 API](#) 取得[交易狀態]，再依[交易狀態]發動欲執行動作。

- 請款使用說明:

1. 信用卡付款成功，查詢訂單會顯示[已授權] (此時僅會佔用信用卡額度)，如果此筆訂單要送至銀行請款，須執行[關帳](Action=C)。
2. 執行[關帳]後，查詢訂單會顯示[要關帳]，表示綠界將要向銀行進行請款作業。
3. 當綠界已向銀行完成[請款]作業，查詢訂單會顯示[已關帳]。

- 退款使用方式:

1. 呼叫[查詢信用卡單筆明細紀錄 API](#) 取得狀態
2. 查詢後，呼叫信用卡請退款 API:
 - [已授權]階段: 執行[放棄] (Action=N)可釋放信用卡佔額。
 - [要關帳]階段:
 - i. 全額退款: 先執行[取消] (Action=E)，接著進行[放棄] (Action=N)。
 - ii. 部份退款: 執行[退刷] (Action=R)。
 - [已關帳]階段: 執行[退刷] (Action=R)。

- 執行動作說明:

- 「關帳」：訂單在每日 20:00 系統會自行關帳，綠界依關帳後的金額向銀行請/退款。

注意事項:

- (1) 若於廠商後台將系統自動關帳關閉以後，訂單仍須於 21 天內完成關帳作業，若無，後續將無法以 API 方式進行關帳，錯誤訊息-error_overDAY 須洽客服手動關帳；若未對訂單進行關帳，第 80 天時將發送信用卡未完成關帳通知；超過 90 天系統會將未關帳訂單進行放棄作業。
- (2) 如付款方式為銀聯卡，關帳方式為立即關帳。

- 「退刷」：在系統關帳或自行關帳後，可修改訂單金額。

※修改上限為訂單金額

注意事項：

- (1) 分期的交易一定得全額退刷，一般授權才能作部份金額退刷。
- (2) 使用紅利折抵交易得全額退刷，僅一般交易才可退刷部分金額。
- (3) 如帳戶餘額低於退刷金額，將無法退刷，建議留存一定金額於綠界帳戶或至廠商後台預存綠界帳戶餘額供退刷之用。

- 「取消」：可取消關帳，讓訂單回復關帳前的狀態。
- 「放棄」：適用在當日關帳前放棄此筆交易，不做請款。

注意事項：

- (1) 此 API 不支援定期定額「停用收款」功能，若要停用定期定額收款請登入廠商後台進行操作，系統路徑為信用卡收單>定期定額查詢>明細 / 編輯。
- (2) 不支援部分金額退款，一律會以訂單全額進行退款。

● 介接路徑：

- 正式環境：<https://ecpayment.ecpay.com.tw/1.0.0/Credit/DoAction>
- 測試環境：因無法提供實際授權，故無法使用此 API

● 廠商傳入參數說明：(JSON 格式)

- Content Type : application/json
- HTTP Method : POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
*Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項： 請留存此欄位編號，之後如需與綠界客服人員追查 API 相關問題，請以此編號作為查詢依據。
*Revision	串接版號	String(10)	請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料		加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RqHeader": {
    "Timestamp": 1525168923,
    "Revision": "2.0.0"
  },
  "Data": "..."
}
```

- Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	平台商代號	String(10)	平台商特店編號
*MerchantID	特店編號	String(10)	綠界提供
*MerchantTradeNo	特店交易編號	String(20)	不可重覆
*TradeNo	綠界交易編號	String(20)	
*Action	執行方式	String(1)	針對訂單做處理的動作，例如要對訂單做關帳時，請帶 C。 C:關帳 R:退刷 E:取消 N:放棄
*TotalAmount	金額	Int	

Data 範例：

```
{
  "MerchantID": "3002607",
  "MerchantTradeNo": "20180914001",
  "TradeNo": "1809261503338172",
  "Action": "C",
  "TotalAmount": 100
}
```

- 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	
RpHeader	回傳資料	JSON	
Timestamp	回傳時間	Number	時間戳 Unix timestamp
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RpHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料		加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "成功",
  "Data": "..."
}
```

- Data 參數說明：請在解密後對參數值進行 URLDecode

參數	參數名稱	型態	說明
RtnCode	執行方式	Int	1 代表 API 執行成功，其餘代碼均為失敗。 失敗代碼請參考 交易訊息代碼表
RtnMsg	交易訊息	String(200)	
RtnCode 為 1 時，以下參數才會有值：			
PlatformID	平台商代號	String(10)	平台商特店編號
MerchantID	特店編號	String(10)	
MerchantTradeNo	特店交易編號	String(20)	不可重覆
TradeNo	綠界交易編號	String(20)	

Data 範例：

```
{  
  "MerchantID": "3002607",  
  "MerchantTradeNo": "20180914001",  
  "TradeNo": "1809261503338172",  
  "RtnCode": 1,  
  "RtnMsg": ""  
}
```


12. 定期定額查詢(Server)

- 應用場景：

提供查詢信用卡定期定額明細記錄。

- 介接路徑：

- 正式環境：<https://ecpayment.ecpay.com.tw/1.0.0/Cashier/QueryTrade>

- 測試環境：<https://ecpayment-stage.ecpay.com.tw/1.0.0/Cashier/QueryTrade>

- 廠商傳入參數說明：(JSON 格式)

- Content Type : application/json

- HTTP Method : POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
*Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
*Revision	串接版號	String(10)	請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RqHeader": {
    "Timestamp": 1525168923,
    "Revision": "1.0.0"
  },
  "Data": "..."
}
```

- Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	平台商編號	String(10)	特約合作平台商特店編號
*MerchantID	特店編號	String(10)	
*MerchantTradeNo	特店交易編號	String(20)	

Data 範例：

```
{
  "MerchantID": "3002607",
  "PlatformID": "2000132",
  "MerchantTradeNo": "123486456"
}
```

- 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號
RpHeader	傳輸資料	JSON	
	Timestamp	Number	時間戳 Unix timestamp 注意事項: 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RqHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
  "Data": "..."
}
```

● Data 參數說明：請在解密後對參數值進行 URLDecode

參數		參數名稱	型態	說明
RtnCode		交易狀態	Int	1 為成功，其餘為失敗
RtnMsg		交易訊息	String(200)	
RtnCode 為 1 時，以下參數才會有值：				
PlatformID		特約合作平台商代號	String(10)	
MerchantID		特店編號	String(10)	
OrderInfo		訂單資訊	JSON	此欄位顯示為定期定額的第一筆資料，第二筆之後的資料為 ExecLog
	MerchantTradeNo	特店交易編號	String (20)	
	TradeNo	綠界的交易編號	String (20)	
	PaymentDate	付款時間	String (20)	yyyy/MM/dd HH:mm:ss
	TradeAmt	交易金額	Int	
	PaymentType	付款方式	String (20)	Credit/CVS/ATM/BARCODE
	TradeDate	訂單時間	String (20)	yyyy/MM/dd HH:mm:ss
	ChargeFee	手續費	Number	格式為整數 8 位，小數 2 位
	TradeStatus	交易狀態	String (8)	回傳值： 若為 0 時，代表交易訂單成立未付款 若為 1 時，代表交易訂單成立已付款
CardInfo		信用卡授權資訊	JSON	
	AuthCode	銀行授權碼	String(6)	
	Gwsr	授權交易單號	Int	
	ProcessDate	交易時間	String(20)	yyyy/MM/dd HH:mm:ss
	Amount	金額	Int	
	Eci	3D(VBV) 回傳值	Int	Eci=5,6,2,1 代表該筆交易為 3D 交易
	Card6No	信用卡卡號前六碼	String(6)	
	Card4No	信用卡卡號末四碼	String(4)	
	PeriodType	週期種類	String(1)	
	Frequency	執行頻率	Int	
	ExecTimes	執行次數	Int	
	PeriodAmount	訂單建立時的每次要授權金額	Int	
	TotalSuccessTimes	已執行成功次數	Int	目前已成功授權的次數。

	TotalSuccessAmount	目前已成功授權的金額合計	Int	
	ExecStatus	執行狀態	String(1)	回傳值如下： 1.若為 0 時代表已取消 2.若為 1 時代表執行中 3.若為 2 時代表執行完成
	ExecLog	每次授權明細		每一次的授權紀錄。 例如：若已成功授權 3 次(TotalSuccessTimes 為 3)，則會顯示 3 筆授權紀錄。
	RtnCode	交易狀態	Int	若回傳值時為 1 時代表授權成功，其餘為失敗，失敗代碼請參考交易訊息代碼一覽表
	Amount	授權金額	Int	
	Gwsr	授權交易單號	Int	
	ProcessDate	授權成功處理時間	String(20)	格式為 yyyy/MM/dd HH:mm:ss
	AuthCode	授權碼	String(6)	
	TradeNo	綠界的交易編號	String(20)	請保存綠界的交易編號與特店交易編號 [MerchantTradeNo] 的關聯。
	ChargeFee	手續費	Number	格式為整數 8 位，小數 2 位
	CustomField	自訂欄位	String(200)	提供特店使用客制化欄位 JSON

Data 解密後：

```

{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "MerchantID": "3002607",
  "OrderInfo": {
    "MerchantTradeNo": "20180914001",
    "TradeNo": "1809261503338172",
    "PaymentDate": "2018/09/26 14:59:54",
    "TradeAmt": "500",
    "PaymentType": "Credit",
    "TradeDate": "2018/09/26 14:59:54",
    "ChargeFee": 10,
    "TradeStatus": "1"
  },
  "CardInfo": {

```

```
"AuthCode": "777777",
"Gwsr": 10735183,
"ProcessDate": "2018/09/26 14:59:54",
"Amount": 100,
"Eci": 2,
"Card4No": "2222",
"Card6No": "491122",
"PeriodType": "",
"Frequency": 2,
"ExecTimes": 1,
"PeriodAmount": 2,
"TotalSuccessTimes": 3,
"TotalSuccessAmount": 5
},
"ExecStatus": "1",
"ExecLog": {
    "RtnCode": "",
    "Amount": 500,
    "Gwsr": "12345",
    "ProcessDate": "2020/12/31 15:12:20",
    "AuthCode": "123456",
    "TradeNo": "12345649781",
    "ChargeFee": 10
}
```

13.查詢 ATM/CVS/BARCODE 取號結果(Server)

- 應用場景：

提供查詢 ATM/CVS/BARCODE 取號結果。

- 介接路徑：

- 正式環境：<http://ecpayment.ecpay.com.tw/1.0.0/Cashier/QueryPaymentInfo>
- 測試環境：<http://ecpayment-stage.ecpay.com.tw/1.0.0/Cashier/QueryPaymentInfo>

- 廠商傳入參數說明：(JSON 格式)

- Content Type : application/json
- HTTP Method : POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
*Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
*Revision	串接版號	String(10)	請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RqHeader": {
    "Timestamp": 1525168923,
    "Revision": "1.0.0"
  },
  "Data": "..."
}
```

- Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	平台商編號	String(10)	特約合作平台商特店編號
*MerchantID	特店編號	String(10)	
*MerchantTradeNo	特店交易編號	String(20)	

Data 範例：

```
{
  "MerchantID": "3002607",
  "PlatformID": "2000132",
  "MerchantTradeNo": "123486456"
}
```

- 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號
RpHeader	傳輸資料	JSON	
	Timestamp	Number	時間戳 Unix timestamp 注意事項: 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RqHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息
Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
  "Data": "..."
}
```

● Data 參數說明：請在解密後對參數值進行 URLDecode

參數	參數名稱	型態	說明
RtnCode	交易狀態	Int	1 為成功，其餘為失敗
RtnMsg	交易訊息	String(200)	
RtnCode 為 1 時，以下參數才會有值：			
PlatformID	特約合作平台商代號	String(10)	
MerchantID	特店編號	String(10)	
OrderInfo	訂單資訊	JSON	
MerchantTradeNo	特店交易編號	String (20)	
TradeNo	綠界的交易編號	String (20)	
PaymentDate	付款時間	String (20)	yyyy/MM/dd HH:mm:ss
TradeAmt	交易金額	Int	
PaymentType	付款方式	String (20)	Credit/CVS/ATM/BARCODE
TradeDate	訂單時間	String (20)	yyyy/MM/dd HH:mm:ss
ChargeFee	手續費	Number	格式為整數 8 位，小數 2 位
TradeStatus	交易狀態	String (8)	回傳值： 若為 0 時，代表交易訂單成立未付款 若為 1 時，代表交易訂單成立已付款
CVSInfo	超商代碼資訊	JSON	
PaymentNo	繳費代碼	String(14)	
ExpireDate	繳費期限	Int	
PaymentURL	繳費連結	String(100)	由綠界科技提供手機上顯示的三段式繳費條碼網頁
BarcodeInfo	超商條碼資訊	JSON	
ExpireDate	繳費期限	String(20)	格式為 yyyy/MM/dd HH:mm:ss
Barcode1	條碼第一段號碼	String(20)	格式為 9 碼數字;如果是代碼，則此欄位回傳空白
Barcode2	條碼第二段號碼	String(20)	格式為 16 碼數字;如果是代碼，則此欄位回傳空白
Barcode3	條碼第三段號碼	String(20)	格式為 15 碼數字;如果是代碼，則此欄位回傳空白
ATMInfo	ATM 資訊	JSON	
BankCode	繳費銀行代碼	String (3)	
vAccount	繳費虛擬帳號	String (16)	
ExpireDate	繳費期限	String (10)	格式為 yyyy/MM/dd
CustomField	自訂欄位	String(200)	提供特店使用客制化欄位

Data 解密後：

```
{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "MerchantID": "3002607",
  "OrderInfo": {
    "MerchantTradeNo": "20180914001",
    "TradeNo": "1809261503338172",
    "PaymentDate": "2018/09/26 14:59:54",
    "TradeAmt": "500",
    "PaymentType": "Credit",
    "TradeDate": "2018/09/26 14:59:54",
    "ChargeFee": 10,
    "TradeStatus": "1"
  },
  "CVSInfo": {
    "PaymentNo": "12345678901234",
    "ExpireDate": 7,
    "PaymentURL": "https://aa.aa.aa"
  },
  "BarcodeInfo": {
    "ExpireDate": "2020/12/31 15:12:20",
    "Barcode1": "123456789",
    "Barcode2": "1234567890123456",
    "Barcode3": "123456789012345"
  },
  "ATMInfo": {
    "BankCode": "800",
    "vAccount": "1234567890",
    "ExpireDate": "2020/12/31"
  }
}
```

14. 信用卡定期定額訂單作業(Server)

- 應用場景：

提供特店執行信用卡定期定額訂單交易作業。

- ReAuth：最新一筆定期定額交易授權失敗時，可發動進行補授權作業。

- Cancel：停用本筆定期定額後續授權交易作業。

- 介接路徑：

- 正式環境：<http://ecpayment.ecpay.com.tw/1.0.0/Cashier/CreditCardPeriodAction>

- 測試環境：<http://ecpayment-stage.ecpay.com.tw/1.0.0/Cashier/CreditCardPeriodAction>

- 廠商傳入參數說明：(JSON 格式)

- Content Type ： application/json

- HTTP Method ： POST

參數	參數名稱	型態	說明
*MerchantID	特店編號	String(10)	特店編號 若為平台商時，參數請帶平台商所綁的特店編號
*RqHeader	傳輸資料	JSON	
*Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項： 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
*Revision	串接版號	String(10)	請參考文件封面的 文件版號 。Ex: 1.0.0
*Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明

範例:

```
{  
  "MerchantID": "3002607",  
  "RqHeader": {  
    "Timestamp": 1525168923,  
    "Revision": "1.0.0"  
  }  
}
```

```

},
"Data": "...
}

```

- Data 參數說明：請在加密前對參數值進行 URLEncode

參數	參數名稱	型態	說明
PlatformID	平台商編號	String(10)	特約合作平台商特店編號
*MerchantID	特店編號	String(10)	
*MerchantTradeNo	特店交易編號	String(20)	
*Action	作業方式	String(20)	ReAuth：補授權失敗交易 Cancel：停用定期定額後續交易

注意事項:

- (1) 定期訂額信用卡補授權作業，限當筆交易最新一筆授權失敗時才能使用，例如第二期授權失敗，但第三期成功時，此時無法補授權第二期失敗的交易。
- (2) 停用交易成功後，無法重新啟用，只能重新發動新定期定額訂單進行交易

Data 範例：

```

{
  "MerchantID": "3002607",
  "PlatformID": "2000132",
  "MerchantTradeNo": "123486456",
  "Action": "ReAuth"
}

```

- 綠界回傳參數說明：(JSON 格式)

參數	參數名稱	型態	說明
MerchantID	特店編號	String(10)	特店編號
RpHeader	傳輸資料	JSON	
Timestamp	傳輸時間	Number	時間戳 Unix timestamp 注意事項: 若時間戳跟綠界伺服器接收到時間超過 10 分鐘時，交易會失敗無法進行
TransCode	回傳代碼	Int	1 代表傳輸資料(MerchantID, RpHeader, Data)接收成功，其餘均為失敗
TransMsg	回傳訊息	String(200)	回傳訊息

Data	加密資料	String	加密過 JSON 格式的資料。 加密方法說明
------	------	--------	--

範例:

```
{
  "MerchantID": "3002607",
  "RpHeader": {
    "Timestamp": 1525169058
  },
  "TransCode": 1,
  "TransMsg": "Success",
  "Data": "..."
}
```

- Data 參數說明：請在解密後對參數值進行 URLDecode

參數	參數名稱	型態	說明
RtnCode	交易狀態	Int	1 為成功，其餘為失敗
RtnMsg	交易訊息	String(200)	
RtnCode 為 1 時，以下參數才會有值：			
PlatformID	特約合作平台商代號	String(10)	
MerchantID	特店編號	String(10)	
MerchantTradeNo	特店交易編號	String(20)	

Data 解密後：

```
{
  "RtnCode": 1,
  "RtnMsg": "Success",
  "MerchantID": "3002607",
  "MerchantTradeNo": "123456789014"
}
```

附錄 1. 參數加密方式說明

依提供 AES 加解密用的 Key 及 IV，請將要加密的資料做 URL Encode 編碼，再進行 AES 加密
AES 加密的強度設定方式是 128 bit，CipherMode：CBC，PaddingMode:PKCS7

- 加密範例：Key=A123456789012345，IV=B123456789012345

(1)加密前 Data 資料：

```
{"Name":"Test","ID":"A123456789"}
```

(2)URLEncode 編碼後結果：

```
%7B%22Name%22%3A%22Test%22%2C%22ID%22%3A%22A123456789%22%7D
```

(3)AES 加密後結果：

```
7woM9RorZKAtXJRVccAb0qhHYm+5lnlhBzyfh5EZdNck7PacNsRHgv/Jvp//ajJidqcQcs0UmAgPQVjXQHeziw==
```

- 解密範例：

(1)Data 密文：

```
7woM9RorZKAtXJRVccAb0qhHYm+5lnlhBzyfh5EZdNck7PacNsRHgv/Jvp//ajJidqcQcs0UmAgPQVjXQHeziw==
```

(2)AES 解密結果：

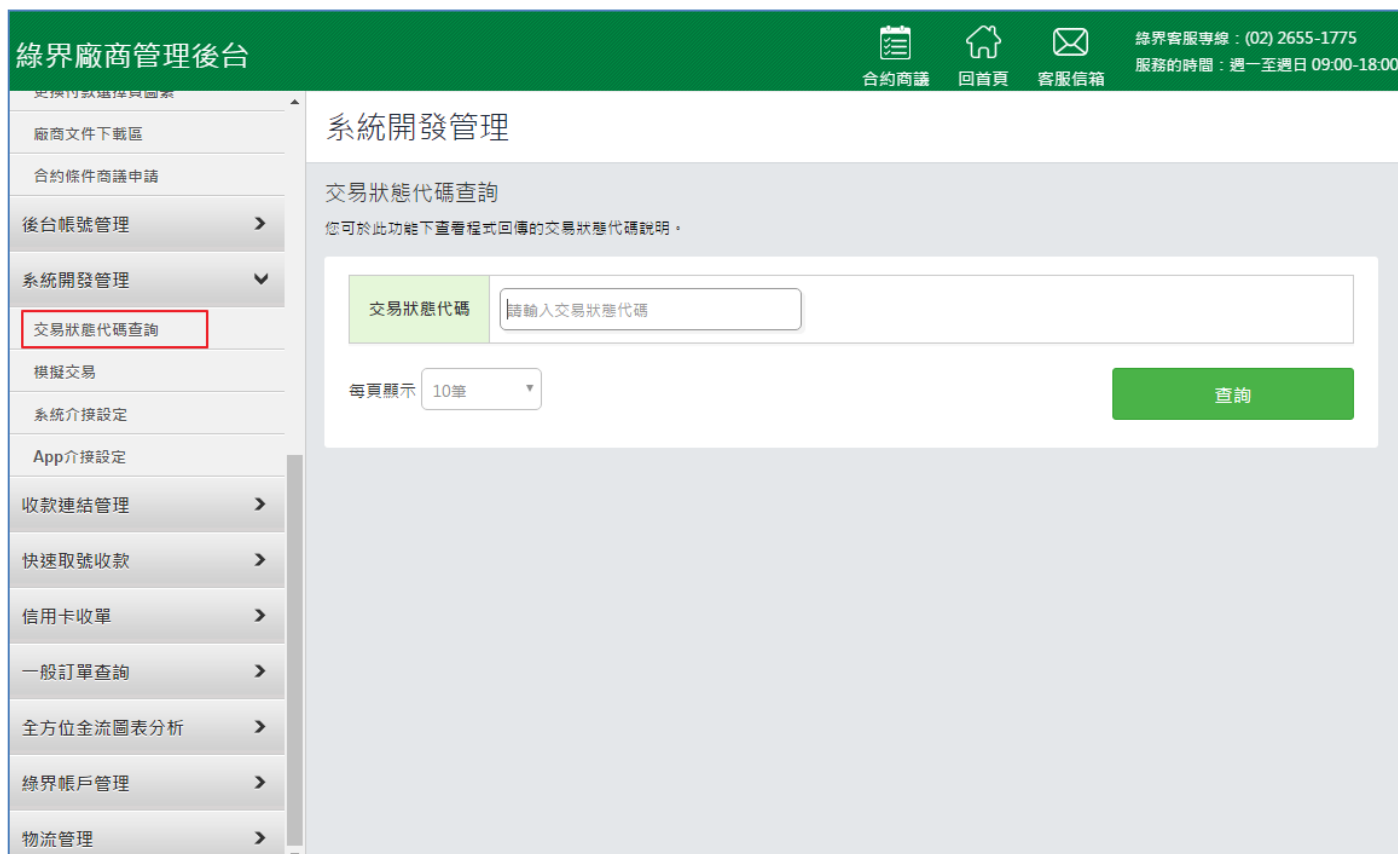
```
%7B%22Name%22%3A%22Test%22%2C%22ID%22%3A%22A123456789%22%7D
```

(3)URLDecode 解碼後結果：

```
{"Name":"Test","ID":"A123456789"}
```

附錄 2. 交易訊息代碼一覽表

因錯誤代碼一直在新增，詳細的錯誤代碼，請到廠商後台->系統開發管理->交易狀態代碼查詢。



The screenshot displays the ECPay Merchant Management Backend interface. The top navigation bar is green and contains the ECPay logo, navigation links (合約商議, 回首頁, 客服信箱), and contact information (綠界客服專線: (02) 2655-1775, 服務的時間: 週一至週日 09:00-18:00). The left sidebar lists various management functions, with '交易狀態代碼查詢' (Transaction Status Code Query) highlighted in red. The main content area is titled '系統開發管理' (System Development Management) and '交易狀態代碼查詢' (Transaction Status Code Query). It includes a description: '您可於此功能下查看程式回傳的交易狀態代碼說明。' (You can view the transaction status code description returned by the program in this function). Below this is a search form with a label '交易狀態代碼' (Transaction Status Code) and a text input field '請輸入交易狀態代碼' (Please enter transaction status code). There is also a '每頁顯示' (Items per page) dropdown menu set to '10筆' (10 items) and a green '查詢' (Query) button.

附錄 3. 銀行代碼表

付款方式	銀行名稱	銀行代碼
ATM	ATM_台新銀行	812
	ATM_玉山(暫不提供)	808
	ATM_台灣銀行	004
	ATM_台北富邦(暫不提供)	813
	ATM_中國信託	822
	ATM_第一銀行	007
	ATM_土地銀行	005
	ATM_國泰世華銀行(暫不提供)	827
	ATM_大眾銀行	814
	ATM_板信銀行	118

附錄 4. Enum 列舉說明

- CallbackStateStatusType/CallbackStatus:

Enum	狀態碼	說明
Fail	0	執行 SDK 失敗;包含網路異常、API 異常等
Success	1	執行 SDK 完成;包含交易付款成功與交易付款失敗皆屬於執行成功。
Cancel	2	取消交易;3D 驗證與銀聯卡驗證頁面關閉，屬於交易中取消交易。
Exit	3	離開付款頁面
Unknown	999	未知且無法預期的錯誤

- PaymentType:

Enum	狀態碼	說明
CreditCard	0	信用卡一次付清
CreditInstallment	1	信用卡分期付款
ATM	2	ATM
CVS	3	超商代碼
Barcode	4	超商條碼
ApplePay	6	Apple Pay
PeriodicFixedAmount	9	信用卡定期定額
UnionPay	11	銀聯卡