



# 著作權

Copyright © LINE Corp. All Rights Reserved.

本文件為 LINE Corp. 之智慧財產,嚴禁違法將本文的部分或全部進行未經授權的拷貝或發放。 本文之目的僅為提供資訊。LINE Corp. 雖期待本文所記載資訊的完整性和正確性,但無法對發生問題的內容之錯誤, 以及疏漏部分負責。因此,使用本文時所產生的責任,皆須由用戶承擔。LINE Corp. 不做任何明示及暗示保證。 包含相關網址在內,本文所言及之軟體商品或商品,皆遵從其所有者所屬地區的著作權法,未遵守相關法律而導致的 所有後果及責任,須由用戶全面承擔。

LINE Corp. 將可能不經預告逕行變更本文之內容。

# 文件資訊

關於本文件

本文件將說明如何開發 LINE Pay 商家提供 LINE Pay 付款的流程。

## 對象

使用本文的對象必須為 LINE Pay 商家伺服器/網站開發者,且理解 HTTP 通訊及 RESTful API。

#### 聯絡資訊

如果您對於本文件內容有任何疑問,或發現任何錯誤,請透過以下方式聯絡我們:

LINE Pay 技術支援 (dl\_lp\_tech\_support@linecorp.com)

#### 修訂記錄

版本	日期	變更記錄
1.0.0	2014.11.10	初稿
1.0.1	2014.12.01	API 之修正、添加附錄
1.0.2	2015.01.15	更新授權、請款 API
1.0.3	2015.01.20	add error code
1.0.4	2015.03.01	OTK 功能、以及 ConfirmUrl 之 Browser 設定功能
		使用 Sandbox 時候的畫面例子
1.0.5	2015.06.01	付款流程更新
		多加 oneTimeKey 說明
		多加 API 變更紀錄
		多加 [附錄] 之 LINE Pay server IP 資訊以及二維碼、條碼測試方法
1.0.6	2015.07.30	全新定義可嘗試重複呼叫之 returnCode
1.1.0	2017.04. 10.	在 Endpoint 及 Header 的 API 中加入新的資訊
		在文件中改變 API 描述的順序
1.1.1	2017.11.28.	修改錯誤範例
1.1.2	2018.01.12	在付款保留 API 的請求參數中·新增額外的欄位·並且修改它的範例。
		- extras.addFriends, extras.branchName

# 文件慣例

原始碼之標記方式

原始碼文字將以灰底黑字的方式呈現。

COPYDATASTRUCT st;

st.dwData = PURPLE\_OUTBOUND\_ENDING;

st.cbData = sizeof(pp); st.lpData = &pp;

::SendMes(GetTargetHwnd(), WM\_COPYDATA, (WPARAM)this->m\_hWnd, (LPARAM)&st);

# 目錄

簡介	9
關於 LINE Pay	
LINE Pay 商家註冊流程	
LINE Pay 功能	
一般付款	
自動付款	
一般功能	
LINE Pay 付款流程	
LINE Pay 整合流程	
一般付款	
自動付款	
如何使用 LINE Pay 的各項 API	
常用特色	
LINE Pay授權	
Request Header	
技術支援	
取得查看付款紀錄 API	
取得查看付款紀錄 Request	
取得查看付款紀錄 Response	
使用取得查看付款紀錄 API 的範例	
付款reserve API	
付款reserve Request	
付款Reserve Response	
使用付款reserve API 的範例	
付款confirm	
付款confirm Request	
付款confirm Response	
使用付款 confirm API 的範例	
退款 API	
退款 Request	
退款 Response	
使用退款 API 的範例	
取得查看授權記錄 API	
取得查看授權記錄 Request	
取得查看授權記錄 Response	

使用取得查看授權記錄 API 的範例	
<i>請款 API</i>	
請款 Request	
請款 Response	
使用請款 API 的範例	
授權作廢 API	
授權作廢 Request	
授權作廢 Response	
使用授權作廢 API 的範例	
自動付款 API	
自動付款 Request	
自動付款 Response	
使用自動付款 API 的範例	
查看 regKey 狀態 API	
查看 regKey 狀態 Request	
查看 regKey 狀態 Response	
使用查看 regKey 狀態 API 的範例	
註銷 regKey API	
註銷 regKey Request	
註銷 regKey Response	
使用註銷 regKey API 的範例	
付録	
LINE Pay 錯誤代碼	
paymentUrl 應用程式使用指南	
對於 Android 應用程式	54
對於 iOS 應用程式	55
confirmUrl 詳細說明以及例外狀況	
confirmUrl 詳細說明	
confirmUrl 例外狀況	
伺服器對伺服器呼叫 confirmUrl 時	
confirmUrl 規格	
使用 confirmUrl 的範例	
Sandbox 使用方法	
Sandbox PC 付款	
Sandbox 手機付款	60
Sandbox OneTimeKey 付款	

# 表單與圖表

## 表單一覽

表 1 檢視查看付款紀錄 Request Parameter	21
表 2 檢視查看付款紀錄 Response Body	21
表 3 付款Reserve Request Parameter	24
表 4 付款Reserve Response Body	27
表 5 付款confirm Request Parameter	30
表 6 付款confirm Response Body	30
表 7 退款 Request Parameter	34
表 8 退款 Response Body	34
表 9 取得查看授權記錄 Request Parameter	36
表 10 取得查看授權記錄 Response Body	36
表 11 請款 Request Parameter	39
表 12 請款 Response Body	39
表 13 授權作廢 Response Body	42
表 14 自動付款 Request Parameter	44
表 15 自動付款 Response Body	44
表 16 查看 regKey 狀態 Request Parameter	47
表 17 查看 regKey 狀態 Response Body	47
表 18 註銷 regKey Response	49
表 19 LINE Pay 錯誤代碼	50
表 20 每API別 ReturnCode總表	52
表 21 confirmUrl Endpoint 規格	57
表 22 confirmURL 參數	57

## 圖表一覽

B	1	LINE Pay 付款流程	_9
며	2	商家註冊流程	_10
며	3	電腦的付款reserve 流程 - 付款畫面	_13
머미	4	手機的付款reserve 流程 - 付款畫面	_14
면	5	電腦的付款畫面 – 付款完成	_15
면	6	手機的付款畫面 - 付款完成	_16
몸	7	伺服器對伺服器呼叫 confirmUrl	17

명	8 oneTimeKey之生成以及 reserve	18
旧	9 自動付款	19
명	10 Sandbox PC 進行付款畫面之例	59
晑	11 Sandbox PC 進行付款畫面之例	60

#### 簡介

# 簡介

本章將說明使用 LINE Pay 的基本概念。

## 關於 LINE Pay

LINE Pay 是一款可供 LINE 用戶在 LINE Pay 商家網站使用的付款系統。



圖 1 LINE Pay 付款流程

當商家註冊成為 LINE Pay 商家後,將可以吸引全世界的 LINE 用戶成為商家的顧客。此外,商家可以透過 LINE 擴展行銷通路,以期提高業績。

想提供 LINE Pay 用戶進行付款的網站必須先註冊為 LINE Pay 商家,才能夠提供用戶使用 LINE Pay 付款。

# LINE Pay 商家註冊流程

一旦商家註冊成為 LINE Pay 商家,將會收到整合金鑰與測試金鑰。商家註冊流程如下圖所示。



圖 2 商家註冊流程

- 前往註冊網頁 (http://pay.line.me). 1.
- 2. 填寫基本資料並送出必要的書面文件
- 3. 審核註冊商家
- 同意手續費及結算週期,然後輸入身分認證碼 4.
- 完成商家註冊 5.
- 傳送完成註冊電子郵件 6.

## LINE Pay 功能

以下為所提供的各項功能。

## 一般付款

一般付款的付款方式是由 LINE Pay 用戶存取 LINE Pay 服務,選擇一種付款方式,然後輸入付款密碼來進行付款。實際商家能夠拿到收款是在商家對此筆交易進行請款後才能向顧客請款。商家可以選擇授權與付款同時發生,就是該筆交易發生同時完成交易。如果商家像鑑賞期之考慮,想要將付款授權與請款兩者分別處理,則應該呼叫「請款 API」。視確認密碼時所選擇的付款方式而定,金額將從買家的餘額或信用卡額度中扣除。.

- 1. 商家伺服器呼叫「付款 reserve API」。
- 2. 買家在 LINE Pay 付款畫面上選擇付款方式並輸入付款密碼。
- 3. 商家伺服器呼叫「付款 confirm API」。
- 4. 付款交易完成。

#### 自動付款

自動付款是在商家伺服器與 LINE Pay 伺服器之間進行,LINE Pay 用戶並未介入。當 LINE Pay 用戶第一次付款時, LINE Pay 會將自動付款的 regKey 核發給商家,然後就像一般付款一樣,經過付款保留與確認流程。此後,商家就可 以使用核發的 regKey,藉由呼叫自動付款 API 來完成以下交易,不需要透過 LINE 應用程式。整個流程如下:

- 1. 第一次付款與核發 regKey (用於自動付款)
  - a. 商家伺服器呼叫付款 reserve API ("payType" :"PREAPPROVED")
  - b. LINE Pay 用戶在 LINE Pay 付款畫面選擇付款方式並輸入付款密碼後,被重新導向到 confirmUrl (付款 reserve API 的參數之一)。
  - c. 商家伺服器呼叫付款 confirm API, 並儲存傳回的 regKey。
  - d. 付款交易完成。
- 2. 自動付款
  - a. 商家伺服器使用 regKey 呼叫自動付款 API。
  - b. 付款交易完成。
- 3. regKey Expiration Request to expire preapproved payment
  - a. 透過 API 終止 regKey
  - b. Expire preapproved payment

#### 一般功能

- 1. 取得詳細付款資料
- 2. 請求退款
- 3. 取得詳細授權資料

4. 取消授權

## LINE Pay 付款流程

在此章節將解釋 LINE Pay 從付款申請到完成付款的流程。



- 1. RESERVE:指呼叫「付款 Reserve API」 時候的初始狀態
- 2. AUTH:指買家在 LINE Pay app 過了選擇付款方式與密碼驗證的步驟
- 3. CANCEL: 指在 LINE Pay app 取消付款
- 4. CAPTURE\_WAIT:指等待商家在後台操作請款(只有 reserve API 的 "capture" 參數為 "false" 時候 才適用
- 5. CAPTURE:指請款
- 6. VOID:指在「CAPTURE\_WAIT」狀態下做取消該筆交易。
- 7. REFUND:指請款後做取消該筆交易

FAILED:指在 LINE Pay API 處理中途遭遇失敗的狀態,在呼叫 confirm, capture, void API 時會發生

簡介

# LINE Pay 整合流程

一般付款

請求付款 - 付款**畫**面

付款 reserve 流程會因 LINE Pay 用戶的使用環境而異,分述如下:

#### 1. 透過電腦



圖 3 電腦的付款 reserve 流程 - 付款畫面

- 1. LINE Pay 用戶從商家的訂購單中選擇一種付款方式。
- 2. 商家呼叫 LINE Pay 的「付款 reserve API」來建立付款 reserve 資訊和取得交易編號及 "paymentUrl" · 其中的交易編號為付款 confirm 時的必要資訊。
- 3. 商家呼叫在步驟 b 中收到的 paymentUrl (由 LINE Pay 提供)。
- 4. LINE Pay 用戶經由通路網頁登入1確認是否為 LINE 用戶。
- 5. 當用戶成功登入 LINE 後, LINE Pay 伺服器推送付款請求到 LINE。
- 6. LINE 用戶在 LINE 應用程式中收到付款請求後,會移動至付款畫面。

<sup>&</sup>lt;sup>1</sup> 這個流程是在使用 LINE 服務前,確認用戶是否為 LINE 用戶。如果商家可以取得 LINE 用戶資訊,則略過通路網頁登入程序,並在付款 reserve 時轉送 mid。

2. 透過手機



圖 4 手機的付款 reserve 流程 - 付款畫面

- 1. LINE Pay 用戶從商家的訂購單中選擇一種付款方式。
- 2. 商家呼叫 LINE Pay 的「付款 reserve API」來建立付款 reserve 資訊和取得交易編號及 "paymentUrl",其中 的交易編號為付款 confirm 時的必要資訊。
- 3. 商家呼叫在步驟 b 中收到的 paymentUrl (由 LINE Pay 提供)。
- 4. 商家使用 paymentUrl 將 LINE Pay 用戶重新導向到 LINE 應用程式中的付款畫面。視整合環境而定, paymentUrl 分成網頁與應用程式。
  - 網頁:由 LINE Pay 提供的網頁畫面。商家先檢查用戶是否已安裝 LINE 應用程式,然後將用戶重新導向 到 LINE Pay 付款畫面。
  - 應用程式:一個應用程式配置 URL,可直接移動到 LINE Pay 付款畫面。有關如何檢查是否已安裝 LINE 應用程式,以及可用的 LINE Pay 版本 (限 Android 平台),請參考 「paymentUrl 應用程式使用指南」

٥

#### 付款畫面 - 付款完成

LINE 用戶選擇 LINE Pay 為付款方式 (信用卡或餘額) 進行付款後, LINE Pay 用戶在完成付款授權會移動到商家在 Reserve 階段收到的 "confirmUrl" (相應的 transactionId 為新增 Parameter, 是為了確認該筆交易)。接著, 用戶到 confirmURL 之後再來商家呼叫 "付款 confirm API", 完成付款。

#### 1. 透過電腦



圖 5 電腦的付款畫面 - 付款完成

- 1. LINE Pay 用戶在 LINE Pay 付款畫面上選擇付款方式並輸入密碼。
- 2. LINE Pay 儲存付款方式資訊,並將付款狀態變更為已授權。
- 3. LINE Pay 用戶檢查付款資訊畫面。
- 4. 當等待付款畫面中的交易變成可付款時,用戶會在付款已預約時被重新導向到從商家處收到的 "confirmUrl"。
- 5. 商家呼叫付款 confirm API · 完成付款。.

#### 2. 透過手機



圖 6 手機的付款畫面 - 付款完成

- 1. LINE Pay 用戶在 LINE Pay 付款畫面上選擇付款方式並輸入密碼。
- 2. LINE Pay 儲存付款方式資訊,並將付款狀態變更為已授權。
- 3. 檢查 LINE 應用程式中的付款資訊畫面後 · LINE Pay 用戶在畫面底部按「確定」 · 在付款已預約時移動到從 商家處收到的 "confirmUrl"。
- 4. 商家呼叫付款 confirm API · 完成付款。

#### 當付款授權與請款兩者分別處理時

整個流程與上述類似,不過商家在呼叫付款 reserve API 時,應該將 "capture" 設定為 "false"。當商家呼叫付款 confirm API,付款狀態會儲存為 "AUTHORIZATION"。

請款時,

- 請款:呼叫請款 API 完成付款。
- 不請款:呼叫授權作廢 API 以取消付款的授權。

## 伺服器對伺服器呼叫 confirmUrl 時

只能藉由商家伺服器與 LINE Pay 伺服器之間的通訊來進行付款 · LINE Pay 用戶未進入商家首頁。



圖 7 伺服器對伺服器呼叫 confirmUrl

- 1. 商家在做付款 reserve 之時,把 confirmUrlType 設定為: "SERVER"。
- 2. LINE Pay 會員進入到 LINE Pay 付款畫面後選擇付款工具,再來輸入密碼。
- LINE Pay 伺服器會儲存付款資訊,做付款 reserve 時再呼叫加盟店傳達的 confirmUrl。
   [附錄] 需參考 "若把 confirmUrl 透過 Server-to-Server 方式傳送"。
- 4. 商家伺服器需要呼叫 confirm API 來做完付款動作。若對於 confirmUrl 的回覆沒有正常傳達之時,就無法呼 叫 confirm API。

若 LINE Pay app 提供之 oneTimeKey 來進行付款

在 LINE Pay app · 為了支援 offline 付款 · 生成 oneTimeKey 。 oneTimeKey 就是替代 LINE 的會員資訊之號碼 · 可以 以條碼或二維碼形式做確認 · 商家可以掃描其代碼獲得 oneTimeKey · 進行付款 ·

前提條件

oneTimeKey 之有效時間:5分鐘(從使用者接近 oneTimeKey 頁面開始計算) oneTimeKey 是與付款 reserve 同時被刪除。

連動流程



圖 8 oneTimeKey 之生成以及 reserve

- 1. 商家要讀取 LINE Pay 的使用者提供的 oneTimeKey (條碼或二維碼)。
- 2. 商家以獲得的 oneTimeKey 的資訊來做付款 reserve。
- 3. 付款 reserve 後, LINE Pay 的會員透過手機的 LINE app 來確認付款邀請。
- 4. LINE Pay 的會員進入 LINE Pay 付款畫面後選擇付款工具,再來輸入密碼。
- 5. LINE Pay 的會員確認完付款資訊頁面後進行付款時,按照商家在 reserve API 設定的 confirmUrlType 參數 而遇到的付款場景會不同。
  - a. CLIENT(預設): confirmUrl 將於買家的瀏覽器打開,買家移動至該 URL 後,商家再呼叫 confirm API 來完成付款動作。
  - b. SERVER: [附錄] 需參考 "若把 confirmUrl 透過 Server-to-Server 方式傳送"。

商家的 Server 呼叫 confirm API 來完成付款流程。若沒有正常收到 confirmUrl 的回覆時,無法呼叫 Confirm API。

## 自動付款

付款請求 - 付款畫面

除了 "payType" 必須設定為 "PREAPPROVED" 之外,整個流程與一般付款相同。

付款畫面 - 付款完成與核發 regKey (用於自動付款)

傳遞 reqKey · 額外附帶付款確認的回應資訊。商家必須儲存此金鑰 · 稍後才能使用自動付款。

自動付款



#### 圖 9 自動付款

當付款確認時,商家伺服器使用 regKey 呼叫 "自動付款 API" 來進行付款。LINE Pay 用戶不介入付款流程,但是在 付款完成時會接到通知。.

#### regKey (用於自動付款) 的註銷

您可以呼叫註銷 regKey API 使不需使用的 regKey 過期。如需詳細資訊,請參閱「註銷 regKey」。

# 如何使用 LINE Pay 的各項 API

本章將詳細說明如何實作 LINE Pay 付款。使用 LINE Pay API 之前,可以在評估流程完成後到商家中心 裡檢視商家 的驗證資訊。

## 常用特色

## LINE Pay 授權

整合 LINE Pay 時所需的驗證資訊如下:

- channel id
- channel secret key

所需的資訊會透過標頭轉送,可以在每個 API 規格中檢視。

## Request Header

項目	資料型別	是否必要?	說明
Content-Type	String	Y	application/json
X-LINE-Channelld	String (10 Bytes)	Y	金流整合資訊 - Channel ID
X-LINE-ChannelSecret	String (32 Bytes)	Y	金流整合資訊 - Channel Secret Key
X-LINE-MerchantDeviceType	String (20 Bytes)	Ν	離線支援 - Device Type

## 技術支援

如果您對於 LINE Pay 整合有任何問題,或發生內部錯誤,請與技術支援 (pay\_tech@linecorp.com) 聯絡。

環境	URL	說明
Beta	https://lgw-pay.line-apps-beta.com	只能在 LINE 內部網路才能測試。
Sandbox	https://sandbox-api-pay.line.me	為了測試連動的環境。於 web simulation page 模擬 LINE Pay App 的付款交易。
Real	https://api-pay.line.me	實際服務營運環境

# 取得查看付款紀錄 API

取得透過 LINE Pay 進行付款的詳細說明。此 API 只會取得已經請款的付款。

## 取得查看付款紀錄 Request

GET /v2/payments

- Connection Timeout:5 秒
- Read Timeout: 20 秒

#### 表 1 檢視查看付款紀錄 Request Parameter

項目	資料型別	是否必要?	說明
transactionId[]	number	Ν	由 LINE Pay 核發的交易編號,用於付款或退款
orderId[]	String	Ν	商家的訂單編號

\* 呼叫此 API 時,需要傳遞一個以上的參數。您最多可以查看 100 筆交易的詳細說明。

## 取得查看付款紀錄 Response

## 表 2 檢視查看付款紀錄 Response Body

項目	資料型別	說明
returnCode	String	結果代碼
	(4byte)	• 0000:成功
		• 1104: 找不到商家
		• 1105: 此商家無法使用 LINE Pay
		• 1106: 標頭資訊錯誤
		• 1150: 找不到交易記錄
		• 1177: 超過允許擷取的交易數目(100)
		• 9000:內部錯誤
returnMessage	String	結果訊息或失敗理由
info[].transactionId	number	交易編號 (19 位數)
info[].transactionDate	String	交易日期與時間 (ISO 8601)
info[].transactionType	String	交易類型
		• PAYMENT:付款

簡介

項目	資料型別	說明
		• PAYMENT_REFUND:退款
		• PARTIAL_REFUND:部分退款
info[].productName	String	產品名稱
info[].merchantName	String	Merchant Name
info[].currency	String	貨幣 (ISO 4217)
Info[].authorizationExpireD ate	String	授權過期日期與時間 (ISO 8601)
info[].payInfo[].method	String	使用的付款方式 (信用卡:CREDIT_CARD.餘額:BALANCE.折扣: DISCOUNT)
info[].payInfo[].amount	number	交易金額 (產生交易編號時的交易金額)
		擷取原始交易時的最終交易金額為
		sum(info[].payInfo[].amount) – sum(refundList[].refundAmount) °
擷取原始交易與發生退款時		
info[].refundList[].refundTra nsactionId	number	退款的交易編號 (19 位數)
info[].refundList[].transactio	String	交易類型
nType		• PAYMENT_REFUND:退款
		• PARTIAL_REFUND:部分退款
info[].refundList[].refundAm ount	number	退款金額
info[].refundList[].refundTra nsactionDate	String	退款的交易日期與時間 (ISO 8601)
擷取退款時		
info[] originalTransactionId	number	

## 使用取得查看付款紀錄 API 的範例

來源 1 取得查看付款紀錄 API 請求

```
GET https://sandbox-api-
pay.line.me/v2/payments?transactionId=20140101123123123&orderId=1002045572 HTTP/1.1
Content-Type: application/json
```

簡介

```
X-LINE-ChannelId: 1414228945
X-LINE-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
X-LINE-MerchantDeviceProfileId: DEVICE_PROFILE_ID
```

來源 2 取得查看付款紀錄 API 回應 (擷取原始交易時)

```
{
   "returnCode":"0000",
   "returnMessage":"success",
   "info":[{
      "transactionId":1020140728100001997,
      "transactionDate":"2014-07-28T09:48:43Z",
      "transactionType":"PAYMENT",
       "payInfo": [
          {"method":"BALANCE", "amount":10},
{"method":"DISCOUNT", "amount":10}
       1,
      "productName":"tes production",
      "currency":"USD",
"orderId":"20140101123123123",
      "refundList":[
          {
              "refundTransactionId":"20213901923123",
             "transactionType":"PARTIAL REFUND",
             "refundAmount":-1,
             "refundTransactionDate":"2014-07-28T09:48:52Z"
          },
          {
             "refundTransactionId":"20213901923123",
             "transactionType":"PARTIAL REFUND",
"refundAmount":-1,
             "refundTransactionDate ":"2014-07-28T09:49:28Z"
          },
          {
              "refundTransactionId":"20213901923123",
              "transactionType":"PARTIAL REFUND",
             "refundAmount":-1,
              "refundTransactionDate ":"2014-07-28T09:53:18Z"
          }
      ]
   }]
3
```

來源 3 取得查看付款紀錄 API 回應 (攝取退款時)

```
'
"returnCode":"0000",
"returnMessage":"success",
"info":[{
    "transactionId":1020140728100001997,
    "transactionDate":"2014-07-28T09:48:43Z",
    "transactionType":"PARTIAL REFUND",
    "amount":-5,
    "productName":"",
    "currency":"USD",
    "orderId":"20140101123123123",
    "originalTransactionId":1020140728100001999
}]
```

## 付款 reserve API

在 LINE Pay 中保留付款資訊。 在使用 LINE Pay 付款前確認是否為正常的商家,然後保留付款所需的資訊。成功付款 reserve 後,商家會收到一個「交易編號」,這個鍵值會在付款完成或退款時使用。

## 付款 reserve Request

## POST /v2/payments/request

- Connection Timeout : 5 秒
- Read Timeout: 20 秒

#### 表 3 付款 Reserve Request Parameter

項目	資料型別	是否必 要?	說明
productName	String (4000 Bytes)	Y	產品名稱 (charset:"UTF-8")
productImageUrl	String (500 Bytes)	Ν	<ul> <li>產品影像 URL (或使用品牌 logo URL)</li> <li>顯示於付款畫面上的影像 URL</li> <li>大小: 84 x 84 (只顯示於付款畫面的影像・ 建議依照指南中的規範製作)</li> </ul>
amount	number2	Y	付款金額
currency	String (3byte)	Υ	付款貨幣 (ISO 4217) 支援的貨幣如下: ・ USD ・ JPY ・ TWD ・ THB
mid	String (50byte)	Ν	LINE 用戶 ID • 將要進行付款的 LINE 使用者之 mid
oneTimeKey	String (12byte)	N	oneTimeKey 是讀取 LINE Pay app 所提供之二維碼、條碼後之結果。替 代 LINE Pay 會員之 mid。有效時間為 5 分鐘,與 rserve 同時會被刪除。

<sup>2</sup> 允許小數位數的數值資料型別

項目	資料型別	是否必 要?	說明
			LINE Pay app 的 QR/BarCode 是從 5.1版本開始支援
confirmUrl Str (50	String (500byte)	Y	買家在 LINE Pay 選擇付款方式並輸入密碼後·被重新導向到商家的 URL。
			• 在重新導向的 URL 上 · 商家可以呼叫付款 confirm API 並完成付款。
			• LINE Pay 會傳遞額外的 "transactionId" 參 數。
			参考
confirmUrlType	String	N	confirmUrl 類型。
	0		買家在 LINE Pay 選擇付款方式並輸入密碼後,被重新導 向到的 URL 所屬的類型。
			• CLIENT: 手機交易流程 (預設)
			<ul> <li>SERVER:網站交易流程。用戶只需要查看 LINE Pay的付款資訊畫面,然後通知商家 伺服器可以付款。</li> </ul>
checkConfirmUrlBrowser	Boolean	Ν	User 前往 confirmUrl 移動時 · 確認使用的瀏覽器相同與 否
			<ul> <li>true: 若買家請求付款的瀏覽器與實際打開 confirmUrl 的瀏覽器不同之時,LINE Pay 將會提供請買家回到原本的瀏覽器的介紹頁 面。</li> </ul>
			• false(預設):不會確認瀏覽器,立即打開 confirmUrl。
			参考
			confirmUrl 詳細說明以及例外狀況
cancelUrl	String	N	取消付款頁面的 URL
	(500byte)		<ul> <li>當 LINE Pay 用戶取消付款後,從 LINE 應 用程式付款畫面重新導向的 URL (取消付款 後,透過行動裝置進入商家應用程式或網站 的商家 URL)。</li> </ul>
			• 商家傳送的 URL 會依現況直接使用。

項目	資料型別	是否必 要?	說明
			• LINE Pay 不會傳遞任何額外的參數。
packageName	String (4000byte)	Ν	在 Android 各應用程式間轉換時 · 防止網路釣魚詐騙的資訊。
orderId	String (100byte)	Y	商家與該筆付款請求對應的訂單編號 <ul> <li>這是商家自行管理的唯一編號。</li> </ul>
deliveryPlacePhone	String (100byte)	Ν	收件人的聯絡資訊 (用於風險管理)
рауТуре	String (12byte)	Ν	<ul><li>付款類型</li><li>NORMAL:單筆付款 (預設)</li><li>PREAPPROVED:自動付款</li></ul>
langCd	String	Ν	<ul> <li>等待付款畫面 (paymentUrl) 的語言代碼。共支援六種語言。</li> <li>ja:日文</li> <li>ko:韓文</li> <li>en:英文</li> <li>zh-Hans:簡體中文</li> <li>zh-Hant:繁體中文</li> <li>th:泰文</li> <li>語言代碼不是強制性的,如果未收到此代碼,則會根據 accept-language 標頭支援多語言。</li> <li>如果收到不支援的 langCd,預設會使用英文 ("en")。</li> <li>BCP-47 格式: http://en.wikipedia.org/wiki/IETF_language</li> </ul>
capture	Boolean	Ν	<ul> <li>指定是否請款</li> <li>true:呼叫付款 confirm API 時・立即進行付款授權與請款(預設)。</li> <li>false:呼叫付款 confirm API 時・只有經過授權・然後透過呼叫 "請款 API" 分開請款・才能完成付款。</li> </ul>

項目	資料型別	是否必 要?	說明
extras.addFriends	object[]	Ν	加好友清單 - type: 服務類型 - "LINE_AT": line@ - idList: ID 清單(在商家中心完成註冊的 LINE@/LINE 官方 帳號清單)
			"addFriends" : [{ "type" : "LINE_AT", "idList" : ["@aaa", "@bbb"] }]
extras.branchName	String (200byte)	Ν	需要付款的商店/分店名稱(僅會顯示前 100 字元)

## 付款 Reserve Response

## 表 4 付款 Reserve Response Body

項目	資料型別	說明	
returnCode	String	結果代碼	
	(4byte)	•	0000:成功
		•	1104:找不到商家
		•	1105:此商家無法使用 LINE Pay
		•	1106:標頭資訊錯誤
		•	1124:金額錯誤 (scale)
		•	1133: 非有效之 oneTimeKey。
		•	1145:正在進行付款
		•	1172: 已有同一訂單編號的交易資料
		•	1178:不支援的貨幣
		•	1194:此商家無法使用自動付款
		•	2101: 參數錯誤
		•	2102: JSON 資料格式錯誤
		•	9000:內部錯誤
returnMessage	String	結果訊息或失敗	救理由。範例如下: 
		•	商家未獲授權・無法付款

項目	資料型別	說明	
		<ul> <li>商家!</li> </ul>	臉證資訊錯誤
info.transactionId	Number	交易編號 (19 位數)	
info.paymentUrl.web	String	付款請求後所前往的	網頁 URL
		• 在網頭	頁交易環境中請求付款時使用
		• LINE	Pay 等待付款畫面的 URL
		• 重新著	尊向到提供的 URL·不附带任何額外參數
		• 在桌	幾打開跳窗時・ 其大小為:Width: 700px,
		Heigł	nt : 546px
info.paymentUrl.app	String.	前往付款畫面的應用	程式 URL
		• 在機图	交易請求付款時使用
		• 將 UI	RL 從商家手機交易重新導向到 LINE 應用
		程式	
		PaymentUrl 應用程	弐使用指南
info.paymentAccessToken String		在 LINE Pay app 輸	入代碼來代替適用掃描器(Scanner)之時候所
		適用的代碼值(共121	位數)。
		(LINE Pay app 的掃	描器是從 5.1 版本開始支援)

## 使用付款 reserve API 的範例

```
來源 4 付款 reserve API 請求
```

```
POST https://sandbox-api-pay.line.me/v2/payments/request HTTP/1.1
Content-Type: application/json
X-LINE-ChannelId: 1414228945
X-LINE-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
X-LINE-MerchantDeviceProfileId: DEVICE PROFILE ID
{
    "productName": "test product",
    "productImageUrl": "http://testst.com",
    "amount": 10,
"currency": "USD",
    "mid": "os89dufgoiw8yer9021384rdfeq",
    "orderId": "20140101123456789",
"confirmUrl":"naversearchapp://inappbrowser?url=
http%3A%2F%2FtestMall.com%2FcheckResult.nhn%3ForderId%3D20140101123456789",
    "cancelUrl":"naversearchapp://inappbrowser?url=
http%3A%2F%2FtestMall.com%2ForderSheet.nhn%3ForderId%3D20140101123456789",
    "capture": "true",
"confirmUrlType": "CLIENT",
"extras" : {
            "addFriends" : [
```

來源 5 付款 reserve API 回應

簡介

# 付款 confirm

此 API 可讓商家完成付款。商家必須呼叫付款 confirm API.才能實際完成付款。不過.當付款 reserve 的 "capture" 參數為 "false" 時.付款狀態會變為 "AUTHORIZATION".只有在呼叫 "請款 API" 後才能完成付款。

## 付款 confirm Request

## POST /v2/payments/{transactionId}/confirm

- Connection Timeout : 5 秒
- Read Timeout: 40 秒

## 表 5 付款 confirm Request Parameter

項目	資料型別	是否必要?	說明
amount	number	Y	付款金額
currency	String	Y	付款貨幣 (ISO 4217)
	(3byte)		支援的貨幣如下。
			• USD
			• JPY
			• TWD
			• THB

## 付款 confirm Response

表 6 付款 confirm Response Body

項目	資料型別	說明	
returnCode	String	結果代碼	
	(4byte)	•	0000:成功
		•	1101: 買家不是 LINE Pay 會員
		•	1102: 買方被停止交易
		•	1104: 找不到商家
		•	1105: 此商家無法使用 LINE Pay
		•	1106: 標頭資訊錯誤
		•	1110: 無法使用的信用卡
		•	1124: 金額錯誤 (scale)
		•	1141: 付款帳戶狀態錯誤
		•	1142: 餘額不足

簡介

項目	資料型別	說明	
			<ul> <li>1150: 找不到交易</li> </ul>
			• 1152: 已有保存付款
			• 1153: 付款 reserve 時的金額與申請的金額不一致
			• 1159: 無付款申請資訊
			• 1169: 付款 confirm 所需要資訊錯誤(在 LINE Pay)
			• 1170: 使用者帳戶的餘額有變動
			• 1172: 已有同一訂單編號的交易資料
			• 1180: 付款時限已過
			• 1198: 正在處理請求
			• 1199: 內部請求錯誤
			• 1280: 信用卡付款時候發生了臨時錯誤
			• 1281: 信用卡付款錯誤
			• 1282: 信用卡授權錯誤
			• 1283: 因異常交易疑慮暫停交易 · 請洽 LINE Pay 客服確認
			• 1284: 暫時無法以信用卡付款
			• 1285: 信用卡資訊不完整
			• 1286: 信用卡付款資訊不正確
			• 1287:信用卡已過期
			• 1288: 信用卡的額度不足
			• 1289: 超過信用卡付款金額上限
			• 1290: 超過一次性付款的額度
			• 1291: 此信用卡已被掛失
			• 1292: 此信用卡已被停卡
			• 1293: 信用卡驗證碼 (CVN) 無效
			• 1294: 此信用卡已被列入黑名單
			• 1295: 信用卡號無效
			• 1296: 無效的金額
			• 1298: 信用卡付款遭拒
			• 9000: 內部錯誤
eturnMessage	String	結果訊息或	
			• 商家未獲授權·無法付款
			• 商家驗證資訊錯誤
nfo.orderId	String	商家在付款	款 reserve 時傳送的訂單編號
nfo.transactionId	number	付款 rese	erve 後.做為結果所收到的交易編號

項目	資料型別	說明	
info.authorizationEx	String	可選擇的授權過期日 (ISO 8601)	
pireDate		• 當付款狀態為 AUTHORIZATION (capture=false) 時	
info.payInfo[].metho d	String	使用的付款方式 (信用卡:CREDIT_CARD · 餘額:BALANCE · 折扣: DISCOUNT)	
info.payInfo[].amou nt	number	付款金額	
info.payInfo[].credit String		(paytype = Preapproved 之時候) 信用卡暱稱	
CardNickname		• 由 LINE Pay 管理的最初登錄的信用卡名稱。	
		• 如果 LINE Pay 用戶未設定暱稱‧則顯示空白文字 ("")。	
		• 此暱稱可由用戶在 LINE Pay 中請求更改,更改後的名稱	
		不會告知商家。	
info.payInfo[].credit	String	(paytype = Preapproved 之時候) 信用卡品牌	
CardBrand		• VISA	
		• MASTER	
		• AMEX	
		• DINERS	
		• JCB	
info.regKey	String	paytype 為 Preapproved 之時候,以後可直接選用的自動付款金鑰 (15 位數)	

## 使用付款 confirm API 的範例

#### 來源 6 付款 confirm API 請求

```
POST https://sandbox-api-pay.line.me/v2/payments/sdhqiwouehrafdasrqoil23as/confirm HTTP/1.1
```

```
Content-Type: application/json
X-LINE-ChannelId: 1414228945
X-LINE-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
X-LINE-MerchantDeviceProfileId: DEVICE_PROFILE_ID
{
    "amount": 1000,
    "currency": "JPY"
}
```

來源 7 付款 confirm API 回應 (payType 設為正常時的儲值交易)

```
"returnCode": "0000",
"returnMessage": "OK",
```

}

```
"info": {
    "orderId": "order 210124213",
    "transactionId": 20140101123123123,
    "payInfo": [{
        "method": "BALANCE",
        "amount": 10
    }, {
        "method": "DISCOUNT",
        "amount": 10
    }]
}
```

來源 8 付款 confirm API 回應 (payType 設為正常時的點數折抵交易)

來源 9 付款 confirm API 回應 (當 payType 設為 PREAPPROVED)

# 退款 API

請求對 LINE Pay 付款完成的項目進行退款。退款時必須提供 LINE Pay 用戶的付款交易編號,也可以視退款金額進行 部分退款。

## 退款 Request

## POST /v2/payments/{transactionId}/refund

- Connection Timeout : 5 秒
- Read Timeout: 20 秒

#### 表 7 退款 Request Parameter

項目	資料型別	是否必要?	說明
refundAmount	number	Ν	退款金額
			- 如果未傳遞此參數・則全額退款

## 退款 Response

## 表 8 退款 Response Body

項目	資料型別	說明	
returnCode	String	結果代碼	
	(4byte)	•	0000: 成功
		•	1101:買家不是 LINE Pay 會員
		•	1102: 買方被停止交易
		•	1104: 找不到商家
		•	1105: 此商家無法使用 LINE Pay
		•	1106: 標頭資訊錯誤
		•	1124: 金額錯誤 (scale)
		•	1150: 找不到交易記錄
		•	1155: 交易編號不符合退款資格
		•	1163: 因超過退款期限·無法退款
		•	1164: 超過退款額度
		•	1165: 交易已進行退款
		•	1179: 無法處理狀態
		•	1198: 正在處理請求

項目	資料型別	說明
		• 1199: 內部請求錯誤
		• 9000: 內部錯誤
		# 若得到 1900 · 1902 · 1999 之代碼 · 可以再試圖重複
		呼叫。
returnMessage	String	結果訊息或失敗理由
info.refundTransactionId	number	退款的交易編號 (新核發的編號 - 19 位數)
info.refundTransactionDate String		退款的交易日期與時間 (ISO 8601)

## 使用退款 API 的範例

#### 來源 10 退款 API 請求

POST https://sandbox-api-pay.line.me/v2/payments/sdhqiwouehrafdasrqoi123as/refund HTTP/1.1

來源 11 退款 API 回應 (payType 設為正常時的儲值交易)

```
{
    "returnCode": "0000",
    "returnMessage": "success",
    "info": {
        "refundTransactionId": 123123123123,
        "refundTransactionDate": "2014-01-01T06:17:41Z"
    }
}
```

## 取得查看授權記錄 API

取得透過 LINE Pay 授權項目的詳細說明。此 API 只會取得已經授權或授權已經作廢的資料;已經請款的項目則可使用「取得查看付款紀錄 API」來檢視。

## 取得查看授權記錄 Request

## GET /v2/payments/authorizations

- Connection Timeout:5 秒
- Read Timeout: 20 秒

#### 表 9 取得查看授權記錄 Request Parameter

項目	資料型別	是否必要?	說明
transactionId[]	number	Ν	由 LINE Pay 核發的交易編號
orderId[]	String	Ν	商家的訂單編號

\* 呼叫此 API 時,需要傳遞一個以上的參數。您最多可以查看 100 筆交易的詳細說明。

## 取得查看授權記錄 Response

#### 表 10 取得查看授權記錄 Response Body

項目	資料型別	說明
returnCode	String	結果代碼
	(4byte)	• 0000:成功
		• 1104: 找不到商家
		• 1105: 此商家無法使用 LINE Pay
		• 1106: 標頭資訊錯誤
		• 1150: 找不到交易記錄
		• 1177: 超過允許擷取的交易數目 (100)
		• 9000: 內部錯誤
returnMessage	String	結果訊息或失敗理由
info[].transactionId	number	交易編號 (19 位數)
info[].transactionDate	String	交易日期 ( <u>ISO 8601</u> )
info[].transactionType	String	交易類型
		• PAYMENT:付款
		• PAYMENT_REFUND:退款

項目	資料型別	說明
		• PARTIAL_REFUND:部分退款
info[].payStatus	String	
		• AUTHORIZATION:已授權
		• VOIDED_AUTHORIZATION:授權已作廢
		(呼叫了「授權作廢 API」)
		• EXPIRED_AUTHORIZATION:過期的授權
		(LINE Pay 允許的授權期限已過期)
info[].payInfo[].method	String	付款方式
		(信用卡:CREDIT_CARD · 餘額:BALANCE · 折扣:DISCOUNT)
info[].payInfo[].amount	number	交易金額 (建立交易編號時)
		最終交易金額為 sum(info[].payInfo[].amount) –
		sum(refundList[].refundAmount) •
info[].productName	String	產品名稱
info[].currency	String	貨幣 (ISO 4217)
info[].orderId	String	商家的訂單編號
info[].authorizationExpireD ate	String	授權過期日期與時間 ( <u>ISO 8601</u> )

#### 使用取得查看授權記錄 API 的範例

來源 12 取得查看授權記錄 API 請求

```
GET https://sandbox-api-
pay.line.me/v2/payments/authorizations?transactionId=201612312312333401 HTTP/1.1
Content-Type: application/json
X-LINE-ChannelId: 1414228945
X-LINE-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
X-LINE-MerchantDeviceProfileId: DEVICE PROFILE ID
```

來源 13 取得查看授權記錄 API 回應

```
"productName": "tes production",
"currency": "USD",
"orderId": "20140101123123123",
"payStatus": "AUTHORIZATION",
"authorizationExpireDate": "2014-07-28T09:48:43Z"
}]
```

## 請款 API

如果商家呼叫付款 reserve API 時 "capture" 為 "false",則只有呼叫請款 API 後才能完成付款。 confirm API 執行結果只有授權的付款進行請款動作。

## 請款 Request

POST /v2/payments/authorizations/{transactionId}/capture

- Connection Timeout : 5 초
- Read Timeout : 60 초

## 表 11 請款 Request Parameter

項目	資料型別	是否必要?	說明
amount	number	Υ	付款金額
currency	String	Y	付款貨幣 (ISO 4217)
	(3byte)		支援下列貨幣:
			• USD
			• JPY
			• TWD
			• THB

## 請款 Response

表 12 請款 Response Body

項目	資料型別	說明	
returnCode	String	結果代碼	
	(4byte)	•	0000: 成功
		•	1104: 找不到商家
		•	1105: 此商家無法使用 LINE Pay
		•	1106: 標頭資訊錯誤
		•	1150: 找不到交易記錄
		•	1155: 交易編號不符合退款資格
		•	1170: 使用者帳戶的餘額有變動
		•	1172: 已有同一訂單編號的交易資料
		•	1179: 無法處理狀態
		•	1183: 付款金額必須大於 0

項目	資料型別	說明
		• 1184: 付款金額比付款申請時候的金額還大
		• 1198: 正在處理請求
		• 1199: 內部請求錯誤
		• 1280: 信用卡付款時候發生了臨時錯誤
		• 1281: 信用卡付款錯誤
		• 1282: 信用卡授權錯誤
		• 1283: 因異常交易疑慮暫停交易,請洽 LINE Pay 客服確
		言辺
		<ul> <li>1284: 暫時無法以信用卡付款</li> </ul>
		• 1285: 信用卡資訊不完整
		• 1286: 信用卡付款資訊不正確
		• 1287: 信用卡已過期
		• 1288: 信用卡的額度不足
		<ul> <li>1289: 超過信用卡付款金額上限</li> </ul>
		• 1290: 超過一次性付款的額度
		• 1291: 此信用卡已被掛失
		• 1292: 此信用卡已被停卡
		• 1293: 信用卡驗證碼 (CVN) 無效
		• 1294: 此信用卡已被列入黑名單
		• 1295: 信用卡號無效
		• 1296: 無效的金額
		• 1298: 信用卡付款遭拒
		• 9000: 內部錯誤
		#若得到 1199, 1280~1298 之代碼·該筆交易會自動被取消。
returnMessage	String	結果訊息或失敗理由。範例如下:
		• 商家未獲授權·無法付款
		• 商家驗證資訊錯誤
info.orderld	String	商家在付款 reserve 時傳遞的訂單編號
info.transactionId	number	付款 reserve API 回傳之「交易號碼」訊息(共 19 位數)
info.payInfo[].method	String	付款方式
		(信用卡:CREDIT_CARD · 餘額:BALANCE, 折扣: DISCOUNT)
info.payInfo[].amount	number	付款金額

簡介

## 使用請款 API 的範例

來源 14 請款 API 請求

```
POST https://sandbox-api-
pay.line.me/v2/payments/authorizations/sdhqiwouehrafdasrqoil23as/capture HTTP/1.1
Content-Type: application/json
X-LINE-ChannelId: 1414228945
X-LINE-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
X-LINE-MerchantDeviceProfileId: DEVICE_PROFILE_ID
{
    "amount": 1000,
    "currency": "JPY"
}
```

來源 15 請款 API 回應

## 授權作廢 API

將已授權的交易作廢。將先前已授權的付款作廢。已經請款的付款可以藉由使用「退款 API」進行退款。

## 授權作廢 Request

POST /v2/payments/authorizations/{transactionId}/void

- Connection Timeout : 5 초
- Read Timeout : 20 초

#### 授權作廢 Response

表 13 授權作廢 Response Body

項目	資料型別	說明	
returnCode	String	結果代碼	
	(4byte)	•	0000:成功
		•	1101: 買家不是 LINE Pay 會員
		•	1102: 買方被停止交易
		•	1104: 找不到商家
		•	1105: 此商家無法使用 LINE Pay
		•	1106: 標頭資訊錯誤
		•	1150: 找不到交易記錄
		•	1155: 交易編號不符合退款資格
		•	1165: 交易已進行退款
		•	1170: 使用者帳戶的餘額有變動
		•	1198: 正在處理請求
		•	1199: 內部請求錯誤
		•	1900: 暫時錯誤,請稍後再試。
		•	1903: 暫時錯誤,請稍候再試。
		•	1999: 嘗試呼叫的資訊與前一次的資訊不符。
		•	9000: 內部錯誤
		# 考	告得到 1900 · 1903 · 1999 之代碼 · 可以再試圖重複呼叫
			•
returnMessage	String	結果訊息或失	敗理由。範例如下:
		•	商家未獲授權・無法付款
		•	商家驗證資訊錯誤

簡介

使用授權作廢 API 的範例

來源 16 授權作廢 API 請求

```
POST https://sandbox-api-
pay.line.me/v2/payments/authorizations/sdhqiwouehrafdasrqoi123as/void HTTP/1.1
Content-Type: application/json
X-LINE-ChannelId: 1414228945
X-LINE-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
X-LINE-MerchantDeviceProfileId: DEVICE_PROFILE_ID
```

來源 17 授權作廢 API 回應

{	
	"returnCode": "0000",
	"returnMessage": "OK"
}	

## 自動付款 API

當付款 reserve API 的付款類型設定為 PREAPPROVED 時, regKey 會隨付款結果傳回。自動付款 API 會使用 regKey 直接完成付款, 無需使用 LINE 應用程式。

## 自動付款 Request

## POST /v2/payments/preapprovedPay/{regKey}/payment

- Connection Timeout : 5 秒
- Read Timeout: 40 秒

## 表 14 自動付款 Request Parameter

項目	資料型別	是否必要?	說明
productName	String	Y	產品名稱
	(4000byte)		
amount	number	Y	付款金額
currency	String	Y	付款貨幣 (ISO 4217)
	(3byte)		支援的貨幣如下:
			• USD
			• JPY
			• TWD
			• THB
orderld	String	Y	商家與該筆付款請求對應的訂單編號。
	(100byte)		這是商家自行管理的唯一編號。
capture	Boolean	N 請	款與否
			• true(預設): 一次處理授權與請款
			• false: 處理到授權後需要呼叫「請款 API」才能完成
			交易

## 自動付款 Response

表 15 自動付款 Response Body

項目 資料型別 說明

項目	資料型別	說明
returnCode	String (4buta)	結果代碼
	(4byte)	• 0000:成功
		• 1101: 買家不是 LINE Pay 會員
		• 1102: 買方被停止交易
		• 1104: 找不到商家
		• 1105: 此商家無法使用 LINE Pay
		• 1106: 標頭資訊錯誤
		• 1110: 無法使用的信用卡
		• 1124: 金額錯誤 (scale)
		<ul> <li>1141: 付款帳戶狀態錯誤</li> </ul>
		<ul> <li>1154: 自動付款帳戶無法使用</li> </ul>
		• 1172: 已有同一訂單編號的交易紀錄
		• 1190: regKey 不存在
		• 1193: regKey 已過期
		• 1198: 正在處理請求
		• 1199: 內部請求錯誤
		• 1280: 信用卡付款時候發生了臨時錯誤
		• 1281: 信用卡付款錯誤
		• 1282: 信用卡授權錯誤
		• 1283: 因異常交易疑慮暫停交易,請洽 LINE Pay 客服確認
		<ul> <li>1284: 暫時無法以信用卡付款</li> </ul>
		• 1285: 信用卡資訊不完整
		• 1286: 信用卡付款資訊不正確
		• 1287: 信用卡已過期
		• 1288: 信用卡的額度不足
		<ul> <li>1289: 超過信用卡付款金額上限</li> </ul>
		• 1290: 超過一次性付款的額度
		• 1291: 此信用卡已被掛失
		• 1292: 此信用卡已被停卡
		• 1293: 信用卡驗證碼 (CVN) 無效
		• 1294: 此信用卡已被列入黑名單
		• 1295: 信用卡號無效
		• 1296: 無效的金額
		• 1298: 信用卡付款遭拒
		• 9000: 內部錯誤
		# 當出現錯誤代碼 1280 - 1287, 1290 - 1294 時, 將會註銷 regKey 45

0

項目	資料型別	說明	
returnMessage	String	結果訊息或失敗理由	
info.transactionId	number	交易編號 (19 位數)	
info.transactionDate	String	交易日期與時間 ( <u>ISO 8601</u> )	
info.authorizationExpireDa String		授權作廢日期 ( <u>ISO 8601</u> )	
te		• 只有做授權(capture=false)時才要傳	

## 使用自動付款 API 的範例

#### 來源 18 自動付款 API 請求

POST https://sandbox-api-pay.line.me/v2/payments/preapprovedPay/RK123asd213/payment HTTP/1.1

```
Content-Type: application/json
X-Line-ChannelId: 1414228945
X-Line-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
{
    "productName": "Naver music monthly pass",
    "amount": 10000,
    "currency": "JPY",
    "orderId": "testOrd2014121200000001"
```

來源 19 自動付款 API 請求

}

```
{
    "returnCode": "0000",
    "returnMessage": "OK",
    "info": {
        "transactionId": 123123123123,
        "transactionDate": "2014-01-01T06:17:41Z"
    }
}
```

# 查看 regKey 狀態 API

使用自動付款 API 前,檢查 regKey 是否存在。

## 查看 regKey 狀態 Request

## GET /v2/payments/preapprovedPay/{regKey}/check

- Connection Timeout : 5 秒
- Read Timeout : 20 秒

## 表 16 查看 regKey 狀態 Request Parameter

項目	資料型別	是否必要?	說明
creditCardAuth	Boolean	Ν	試圖驗證買家在 regKey 設定之信用卡之最少金額與否
			• true :
			- 驗證 LINE Pay 內部檔案之有效性
			- 驗證信用卡最少金額付款
			- 在商家資訊激活「最少金額驗證功能」才可適
			用·且需要 LINE Pay 管理者之檢驗。
			• false(預設):
			- 只驗證 LINE Pay 內部檔案之有效性.

## 查看 regKey 狀態 Response

表 17 查看 regKey 狀態 Response Body

項目	資料型別	說明	
returnCode	String(4byte)	結果代碼	
		•	0000: 成功
		•	1101: 買家不是 LINE Pay 會員
		•	1102: 買方被停止交易
		•	1104: 找不到商家
		•	1105: 此商家無法使用 LINE Pay
		•	1106: 標頭資訊錯誤
		•	1141: 無法驗證在 regKey 設定之信用卡
		•	1154: 自動付款帳戶無法使用
		•	1190: regKey 不存在

簡介

## LINE Pay 整合指南 – 內部商家用

項目	資料型別	說明
		• 1193: regKey 已過期
returnMessage	String	結果訊息或失敗理由
使用查看 regKey	/ 狀態 API 的範例	

來源 20 查看 regKey 狀態 API 請求

GET https://sandbox-api-pay.line.me/v2/payments/preapprovedPay/RK7862119XTF1TT/check
HTTP/1.1
Content-Type: application/json
X-Line-ChannelId: 1414228945
X-Line-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27

來源 21 查看 regKey 狀態 API 回應

{
 "returnCode": "0000",
 "returnMessage": "OK"
}

# 註銷 regKey API

註銷為自動付款登錄的 regKey 資訊。一旦呼叫此 API,現有的 regKey 將無法繼續用於自動付款。

## 註銷 regKey Request

POST /v2/payments/preapprovedPay/{regKey}/expire

- Connection Timeout:5 秒
- Read Timeout: 20 秒

## 註銷 regKey Response

表 18 註銷 regKey Response

項目	資料型別	說明	
returnCode	String	結果代碼	
	(4byte)	•	0000: 成功
		•	1104: 找不到商家
		•	1105: 此商家無法使用 LINE Pay
		•	1106: 標頭資訊錯誤
		•	1190: regKey 不存在
		•	1193: regKey 已過期
		•	9000: 內部錯誤
returnMessage	String	結果訊息或失敗	收理由

## 使用註銷 regKey API 的範例

#### 來源 22 註銷 regKey API 請求

POST https://sandbox-api-pay.line.me/v2/payments/preapprovedPay/RK123asd213/expire HTTP/1.1

```
Content-Type: application/json
X-Line-ChannelId: 1414228945
X-Line-ChannelSecret: 15cf4348793ea65cae10959b7a29fe27
```

#### 來源 23 註銷 regKey API 回應

}

```
"returnCode": "0000",
"returnMessage": "OK"
```

# 附錄

# LINE Pay 錯誤代碼

本節定義了 LINE Pay 中的錯誤代碼。錯誤代碼的 returnMessage 以英文呈現,如果沒有訊息,則顯示連字號 (-)。

表 19 LINE Pay 錯誤代碼

代碼(returnCode)	說明
1101	買家不是 LINE Pay 會員
1102	買方被停止交易
1104	找不到商家
1105	此商家無法使用 LINE Pay
1106	標頭資訊錯誤
1110	無法使用的信用卡
1124	金額錯誤 (scale)
1133	非有效之 oneTimeKey
1141	付款帳戶狀態錯誤
1142	餘額不足
1145	正在進行付款
1150	找不到交易記錄
1152	已有既存付款
1153	付款 reserve 時的金額與申請的金額不一致
1154	買家設定為自動付款的信用卡變無法使用狀態
1155	交易編號不符合退款資格
1159	無付款申請資訊
1163	可退款日期已過無法退款
1164	超過退款額度
1165	交易已進行退款
1169	付款 confirm 所需要資訊錯誤(在 LINE Pay)
1170	使用者帳戶的餘額有變動
1172	已有同一訂單編號的交易履歷
1177	超過允許擷取的交易數目 (100)

代碼(returnCode)	說明
1178	不支援的貨幣
1179	無法處理狀態
1180	付款時限已過
1183	付款金額必須大於 0
1184	付款金額比付款申請時候的金額還大
1190	regKey 不存在
1193	regKey 已過期
1194	此商家無法使用自動付款
1197	已在處理使用 regKey 進行的付款
1198	正在處理請求
1199	內部請求錯誤
1280	信用卡付款時候發生了臨時錯誤
1281	信用卡付款錯誤
1282	信用卡授權錯誤
1283	因異常交易疑慮暫停交易,請洽 LINE Pay 客服確認
1284	暫時無法以信用卡付款
1285	信用卡資訊不完整
1286	信用卡付款資訊不正確
1287	信用卡已過期
1288	信用卡的額度不足
1289	超過信用卡付款金額上限
1290	超過一次性付款的額度
1291	此信用卡已被掛失
1292	此信用卡已被停卡
1293	信用卡驗證碼 (CVN) 無效
1294	此信用卡已被列入黑名單
1295	信用卡號無效
1296	無效的金額
1298	信用卡付款遭拒

## LINE Pay 整合指南 – 內部商家用

代碼(returnCode)	說明
2101	參數錯誤
2102	JSON 資料格式錯誤
9000	內部錯誤

如果發生內部錯誤 "returnCode:1199" · 請與技術支援 (pay\_tech@linecorp.com) 聯絡

## 表 20 每 API 別 ReturnCode 總表

錯誤代碼	reserve	confirm	請款	授權作廢	退款	授權履歷	付款紀錄	自動付款	查看 regKey	註銷 regKey
1101		V		V	V			V	V	
1102		v		V	V			V	V	
1104	V	V	V	V	v	V	V	V	V	V
1105	V	V	V	V	v	V	V	V	V	V
1106	V	V	V	V	v	V	V	V	V	V
1110		V						V		
1124	V	V			v			V		
1133	V								V	
1141		v						V		
1142		v								
1145	V									
1150			V	V	V	V	V			
1152		V								
1153		v								
1154								V	V	
1155			V	V	v					
1159		V								
1163					V					
1164					V					
1165				V						
1169		v								
1170		v	V	V						
1172	V	V	V					V		
1177						V	V			
1178	V									
1179			V	V	V					

錯誤代碼	reserve	confirm	請款	授權作廢	退款	授權履歷	付款紀錄	自動付款	查看 regKey	註銷 regKey
1180		V								
1183	V		V					V		
1184			V							
1190								V	V	V
1193								V	V	
1194	V									
1198		V	V	V	V			V		
1199		V	V	V	V			V		
2101	V	V	V	V	V	V	V	V	V	V
2102	V	V	V	V	V	V	V	V	V	V
9000	V	V	V	V	V	V	V	V	V	V

由於 1280 至 1298 的錯誤代碼是由信用卡公司回傳的信用卡錯誤,這邊不再另行解釋錯誤原因。

# paymentUrl 應用程式使用指南

本節提供在商家應用程式請求付款後,重新導向至 LINE 應用程式的用法說明。

#### 對於 Android 應用程式

使用以下範例程式碼檢查是否已安裝 LINE 應用程式,以及可用的 LINE Pay 版本。如果已經安裝 LINE 應用程式, 且已檢查可用的 LINE Pay 版本,則將用戶重新導向到 LINE Pay 付款畫面。

```
int linePaySupportedVersion = 230;
String paymentUrl = "..."; // This is "paymentUrl.app" URL String.
Context context = getActivity();
try {
   PackageManager pm = context.getPackageManager();
   PackageInfo packageInfo = pm.getPackageInfo("jp.naver.line.android", 0);
   int versionCode = packageInfo.versionCode;
   if (linePaySupportedVersion <= versionCode)</pre>
       launchUri(paymentUrl);
   } else {
       confirmLineInstall(context);
   }
} catch (NameNotFoundException e) {
   confirmLineInstall(context);
private void confirmLineInstall(Context context) {
   new AlertDialog.Builder(context)
   .setTitle("LINE Pay")
   .setMessage(getString(R.String.linepay confirm))
   .setCancelable(false)
    .setPositiveButton(getString(R.String.linepay install), new
DialogInterface.OnClickListener() {
       @Override
       public void onClick(DialogInterface dialog, int which) {
          launchUri("market://details?id=jp.naver.line.android");
       }
   })
    .setNegativeButton(getString(R.String.linepay cancel), new
DialogInterface.OnClickListener() {
       QOverride
       public void onClick(DialogInterface dialog, int which) {
   })
   .show();
private void launchUri(String uriString) {
   Uri uri = Uri.parse(uriString);
   Intent intent = new Intent(Intent.ACTION VIEW, uri);
   startActivity(intent);
```

附錄

## 對於 iOS 應用程式

使用以下範例程式碼檢查是否已安裝 LINE 應用程式。如果已經安裝 LINE 應用程式,則將用戶重新導向到 LINE Pay 付款畫面。

```
NSString* lineScheme = @"line://";
BOOL installed = [[UIApplication sharedApplication]
               canOpenURL:[NSURL URLWithString:lineScheme]];
if (installed) {
   UTAlertView *alert =
   [[UIAlertView alloc] initWithTitle:@"LINE Pay"
message:NSLocalizedString(@"linepay.confirm", nil)
                         delegate:self
cancelButtonTitle:NSLocalizedString(@"linepay.ok", nil) otherButtonTitles:nil];
   alert.tag = 1;
   [alert show];
} else {
   UIAlertView *alert =
   [[UIAlertView alloc] initWithTitle:@"LINE Pay"
message:NSLocalizedString(@"linepay.confirm", nil)
                         delegate:self
cancelButtonTitle:NSLocalizedString(@"linepay.cancel", nil)
                         otherButtonTitles:NSLocalizedString(@"linepay.install",
nil), nil];
   alert.tag = 2;
   [alert show];
}

    (void)alertView:(UIAlertView*)alertView

clickedButtonAtIndex:(NSInteger)buttonIndex {
   if (alertView.tag == 1 && buttonIndex == 0) {
      NSString *paymentUrl = ...; // This is "paymentUrl.app" URL String.
      [self launchUrl:paymentUrl];
   } else if (alertView.tag == 2 && buttonIndex == 1) {
       [self launchUrl:@"itms-
apps://itunes.apple.com/WebObjects/MZStore.woa/wa/viewSoftware?id=443904275&mt=8"];
   }
}
  (void) launchUrl: (NSString*) urlString {
   NSURL *url = [NSURL URLWithString:urlString];
   [[UIApplication sharedApplication] openURL:url];
```

#### File : en.lproj/Localized.Strings

```
"linepay.confirm" = "Supported by Android/iPhone LINE versions 4.8.0 or higher.";
"linepay.ok" = "OK";
"linepay.cancel" = "Cancel";
"linepay.install" = "Get it now";
```

## confirmUrl 詳細說明以及例外狀況

## confirmUrl 詳細說明

是指使用者在 LINE Pay app 做完付款驗證動作(確認付款工具以及付款密碼)後回去的 callback url。此 url 並不代表付 款動作之完成,呼叫 LINE Pay 所提供的 confirm API 後,最後完成付款動作之商家端的 url。

## confirmUrl 例外狀況

在 LINE Pay · 經過使用者的付款驗證後呼叫 confirmUrl 時 · 隨著使用者的使用環境 · 有可能發生要求付款的出發 Browser 與收到 confirmUrl 的 Browser 不同之狀況 ·



付款 reserve API 的 "CheckConfirmUrlBrowser"是 "true" 時

LINE Pay 會確認 browser 啟動付款的 browser 與移動的 browser 是否一致。由起判斷之結果,

- 若一致:移動到 confirmUrl。
- 若不一致:對使用者介紹並要求移動到啟動付款的 browser。

付款 reserve API 的 "CheckConfirmUrlBrowser" 是 "false(預設值)" 時

若商家有基於 Browser 之 session data(例:登入 session)處理之內容,呼叫 confirmUrl 時有可能發生使用者的付款 無法正常進行。因此,商家連動 LINE Pay 付款時,必須要如下面去呈現。

- 1. 呼叫 confirmUrl 時,不履行基於 session data 的驗證, confirmUrl 包含的資訊來完成付款動作。
- 2. 之後如何處理 session data 之例子
  - a. 若能確認 session data,露出原本商家的付款結果畫面
  - b. 若無法確認 session data,以下兩種方式中選一來呈現。
    - 再生成 session data (例:登入)後露出付款結果畫面
    - 對使用者介紹並要求回到啟動付款的 browser

## 伺服器對伺服器呼叫 confirmUrl 時

本節提供從 LINE Pay 伺服器對商家伺服器呼叫 confirmUrl 的規格與範例。

#### 若商家自己管理 ACL(Access Control List)之時候

商家需要登記並管理 LINE Pay server 之 IP。LINE Pay 在每個環境下的 IP 資訊為如下。

- Sandbox :182.162.196.200
- Real: 211.249.40.1~211.249.40.30

## confirmUrl 規格

表 21 confirmUrl Endpoint 規格

項目	說明
Protocol	HTTP
Method	GET
Request timeout	Connection: 5 秒
	Read: 20 秒

表 22 confirmURL 參數

參數	說明
orderld	商家在付款 reserve 時傳遞的訂單編號
transactionId	預約交易後,做為結果傳回的交易編號

商家不回應對 confirmUrl 的呼叫,因此 LINE Pay 透過檢查 HTTP 狀態代碼來判斷是否成功。如果狀態代碼不是 "200 OK",則通知 LINE Pay 用戶付款未能成功完成。

#### 使用 confirmUrl 的範例

使用 confirmUrl 的範例如下所示,其中商家所提供的 confirmUrl 為 "http://testmall.com/pay/result"。

請求 (LINE Pay -> 商家)

http://testmall.com/pay/result?orderld=2014xxx1232132&trasactionld=201408011234567890

回應

```
// 若成功
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Accept-Charset: utf-8
Content-Type: text/html
Content-Length: 2
Date: Sat, 11 Oct 2014 02:45:48 GMT
// 若失敗
HTTP/1.1 404 NOT_FOUND
Server:Apache-Coyote/1.1
```

Accept-Charset:utf-8 Content-Type:text/html Content-Length:4 Date:Sat, 11 Oct 2014 02:45:48 GMT

# Sandbox 使用方法

在 LINE Pay 的 Sandbox 環境做測試時,需要使用記載在此連動集成 API Endpoint 明細上的 sandbox 来進行付款, 付款画面要由商家自行准备。在 Sandbox 環境試圖付款即将会如下去做處理。

※ 特殊事項

在 Sandbox 環境不管使用電腦或手机都需要登入 Channel Log-in。

(但·在实际环境上連動時·手机付款因為它已透過 LINE 收集使用者的 MID 資訊·不需登入 Chaneel Log-in。)



Sandbox PC 付款

圖 10 Sandbox PC 進行付款畫面之例

## Sandbox 手機付款



圖 11 Sandbox PC 進行付款畫面之例

## Sandbox OneTimeKey 付款

商家可讀取測試人員在 LINE Pay app 適用 "My Code" 來顯示的二維碼或條碼來進行付款。

在 Sandbox 用 web page 形式去提供 "My Code" ·可以測試付款。在 Sandbox 環境下 · 商家讀完 My Code 後再來呼叫 Reserve API · 顯示 My Code 的該頁面會轉移成模擬付款頁面。

- Sandbox My code url : https://sandbox-web-pay.line.me/web/sandbox/payment/otk