TapPay

Apple Pay - Getting Started

Revision History

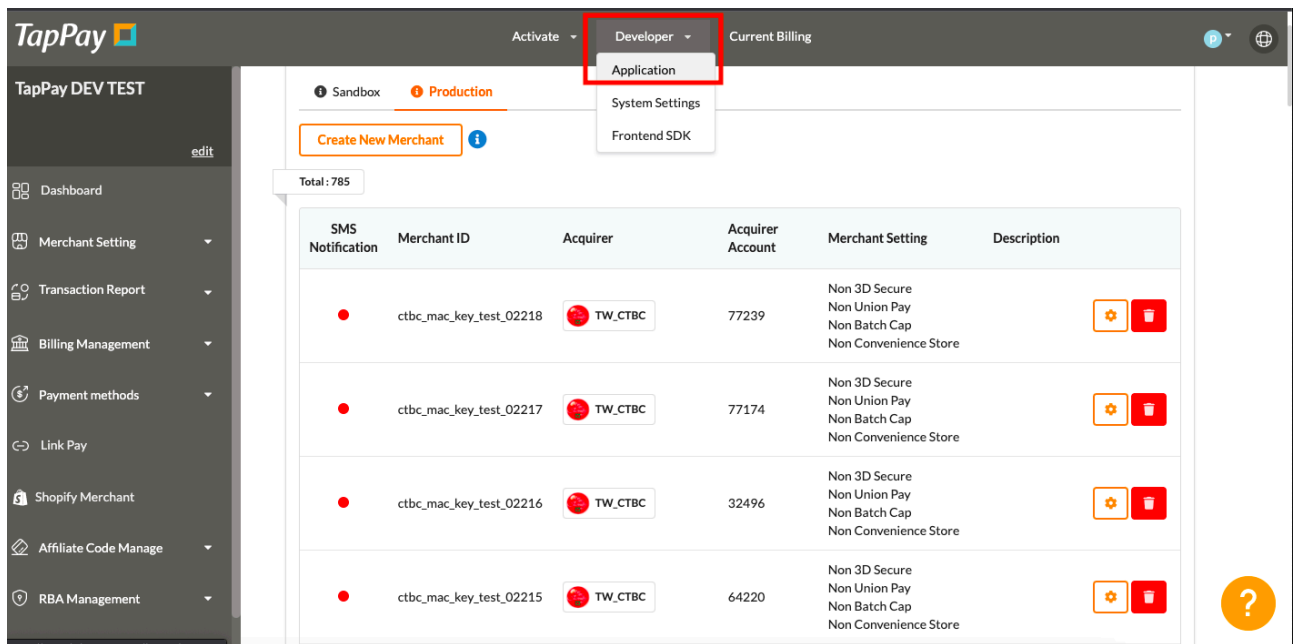| Version	Date	Content
1.0.0	2023.05.12	First Draft

Table of Contents
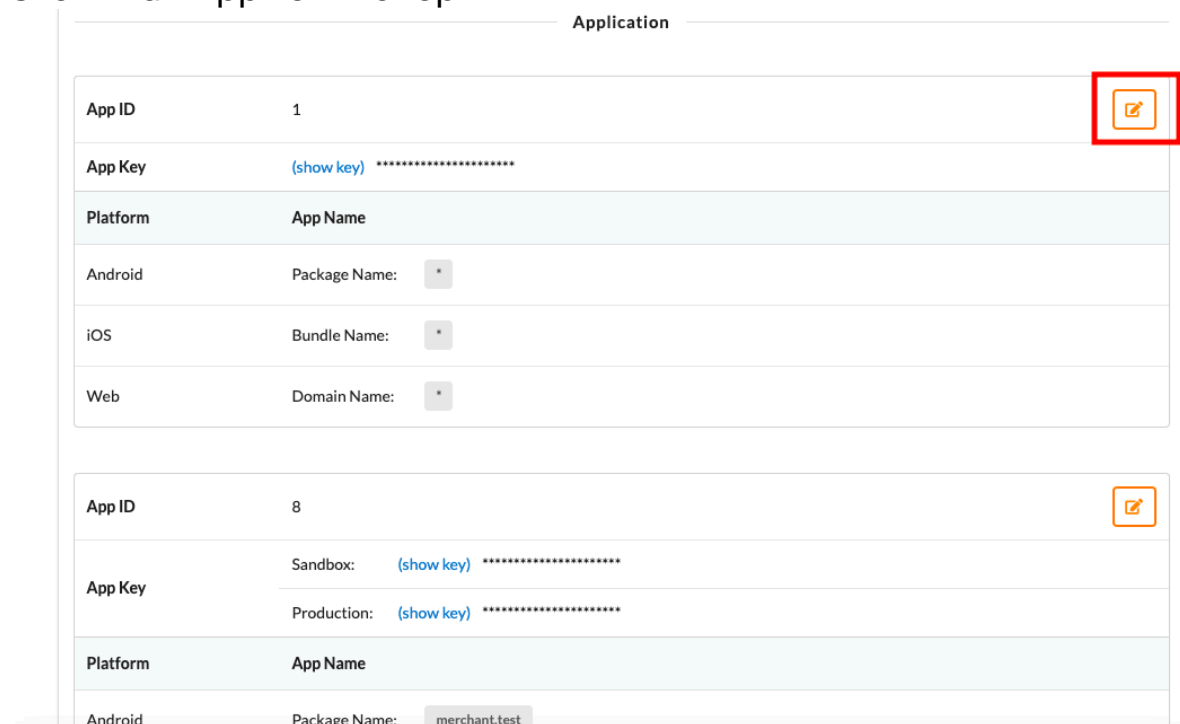
Portal

App Registration

1. Log in to your Portal account.

2. Select "Application" in Developer.

3. Click "Edit App" on the top.

4. Input the appropriate information.

   Bundle name represents the unique identifier of the application. Domain name represents the URL of the website. You may separate multiple domains with semicolons(;).

5. You may view your app ID, app key, and bundle name here.

Merchant Registration

1. Log in to your [Portal account](#).

2. Select "Create Merchant" on the left hand side.
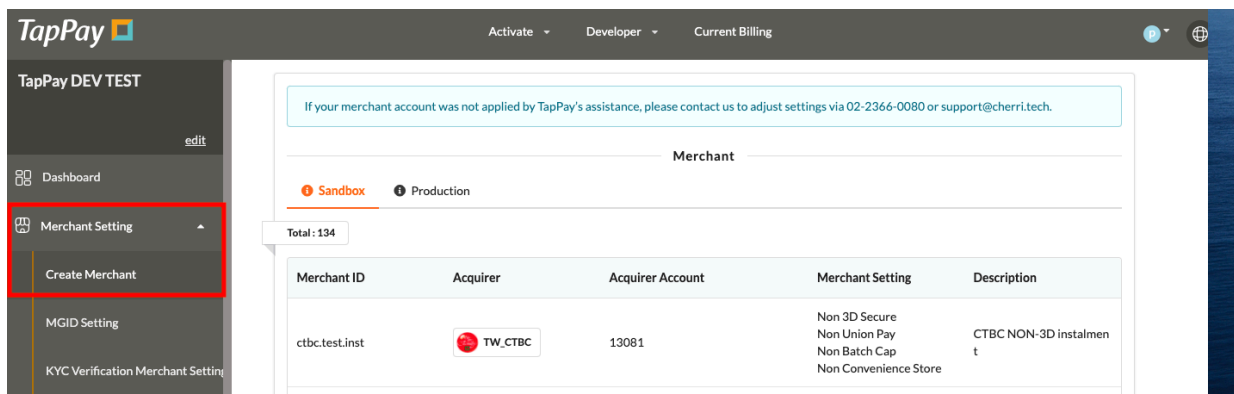
3. Click "Create Merchant" on the top.

   You may not create new merchants in Sandbox environment. Please use the preexisting merchants instead.
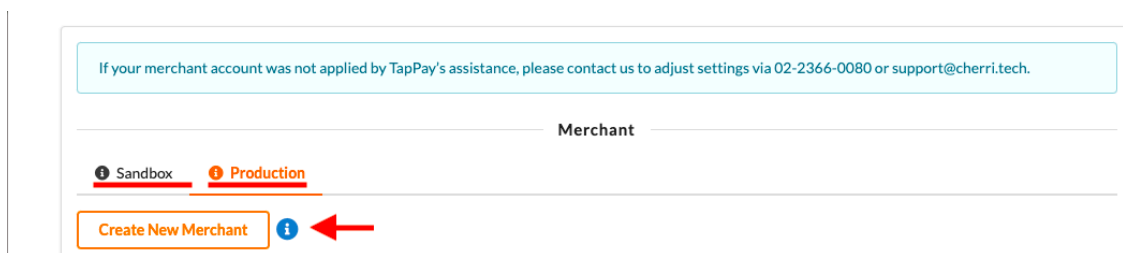
4. Input the appropriate information.

Name	Usage
Connected Acquirer*	Acquirer bank or payment processor.
Acquirer Account*	Acquirer account number.

Create New Merchant

1 Fill the merchant info	**2** Fill the card number	**3** Merchant Test Result

Description

Connected Acquirer *

TAISHIN INTERNATIONAL BANK

☐ **Enable 3D Secure (If enabled, only 3D Secure redirect transaction is supported)**

☐ **Create Sub Merchant**

Merchant ID *
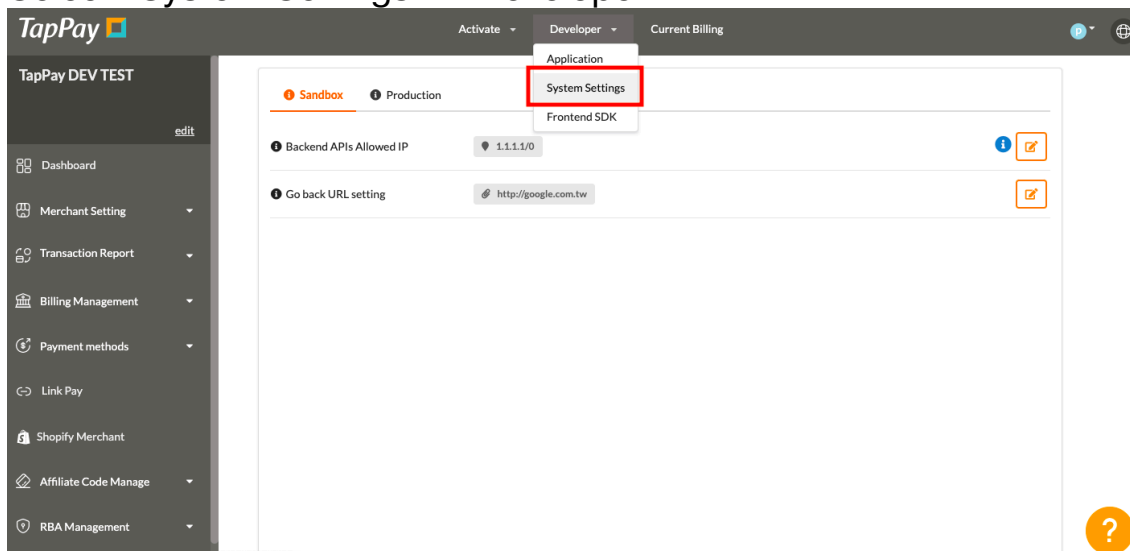
Account ID provided by Acquirer, Bank or Processor

Submit   Cancel

IP Registration
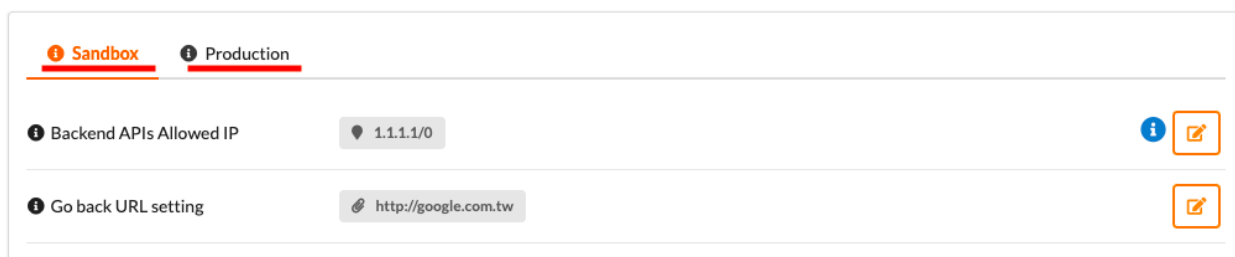
*Please bind your server's IP address to our server first before testing.

1. Log in to your Portal account.

2. Select "System Settings" in Developer.

3. Select the correct environment.

   Make sure you choose the right environment on the top.
   Sandbox environment does **NOT** share the same allowable IPs as production environment.

4. Input your IP.
   This list will bind your server's IP address to our server.
   You may separate multiple IPs with semicolons(;).
   The number after the backslash indicates subnet mask(0~32).
   Each IPv4 address consists of 32 bits: 8bits.8bits.8bits.8bits
   So, the range of all IPv4 addresses, in dot-decimal notation, falls between 0.0.0.0 ~ 255.255.255.255
   The subnet mask determines the number of bits that must match your IP address.

Please refer to the following example:

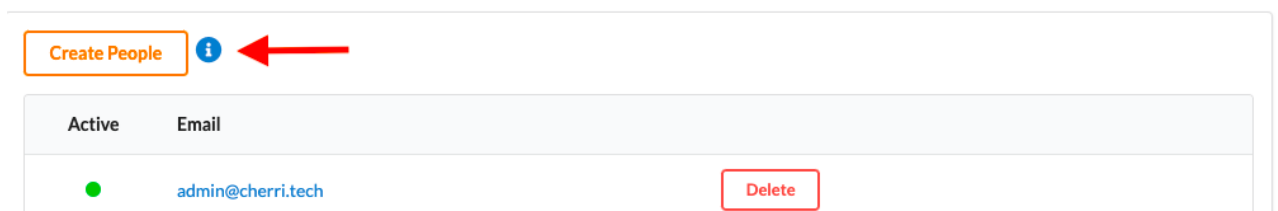| IP	Subnet Mask	Acceptable IP
1.2.3.4/0	0	0.0.0.0 ~ 255.255.255.255
1.2.3.4/8	8	1.0.0.0 ~ 1.255.255.255
1.2.3.4/16	16	1.2.0.0 ~ 1.2.255.255
1.2.3.4/24	24	1.2.3.0 ~ 1.2.3.255
1.2.3.4/32	32	1.2.3.4

People Registration

1. Log in to your Portal account.

2. Select "Account Management" on the top.
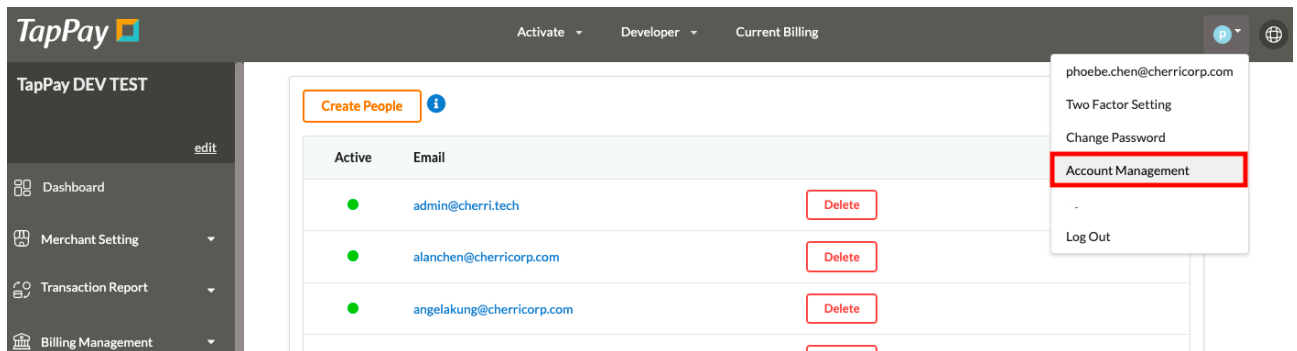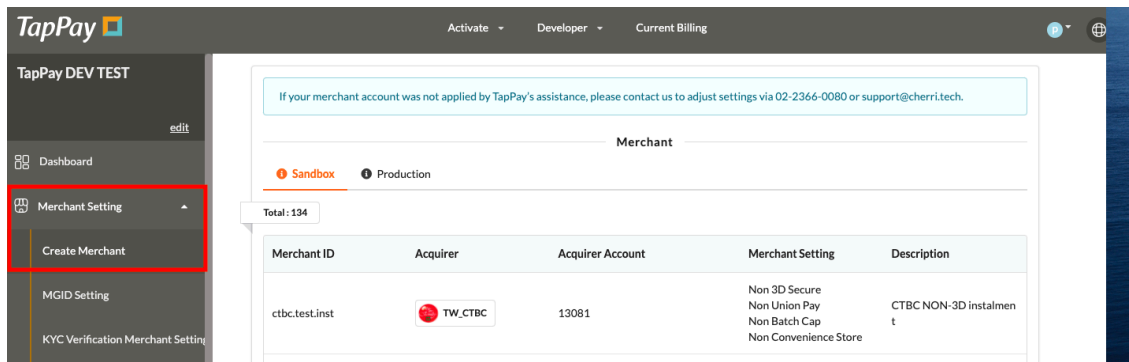

3. Click "Create People".

4. You may register new accounts and alter their permission status.

   You may use our default options or change the permission yourself.

Merchant Notification

1. Log in to your Portal account.

2. Select "Create Merchant" on the left hand side.

3. Select the merchant you wish to enable this feature.

4. Input the appropriate information.

Apple Pay on the Web Merchant Setting

If you enable Apple Pay on the Web Merchant Application Service, you can ignore all the processes in Getting Started > Apple Pay (it means you don't need to register Apple Pay Developer Account).
You just need to set up the relevant information on TapPay Portal and you will get Apple Pay merchant identifier successfully.

Please complete the following steps:

1. Log in to your Portal account.
2. Choose Payment methods > Apple Pay on the Web, click 『Create Apple Pay Setting 』
3. Fill in domain names, download domain verification file, host the domain verification file under the storage path, and then click 『Submit』
4. After finishing settings, you will get an Apple Pay merchant identifier. Please use this merchant identifier in the Sandbox environment.

If you want to enable this service in the Production environment, please complete following steps：
1. Contact TapPay Support
2. Log in to your Portal account.
3. Go to Activate > Apple Pay on The Web. Fill in the website and agree to the declaration. Foundation Approved ID is required if you are a foundation.

4. You will see the following picture after submitting.

5. TapPay will send a notification letter to contact person mailbox after reviewing. And you can enjoy your service in the Production environment!

Apple Pay

Create Merchant ID
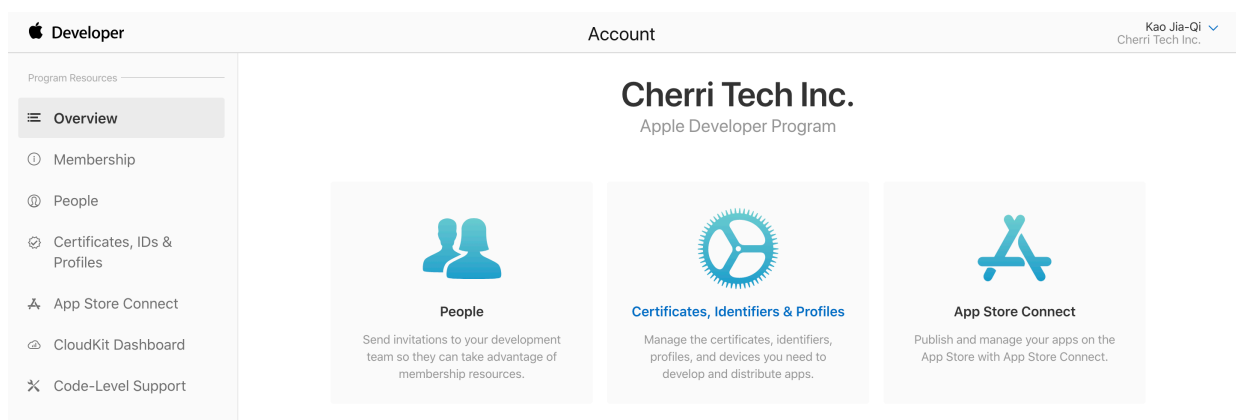
If you meet any condition as below, please complete the following steps.
1. Need a developer account for Web but didn't purchase [Apple Pay on the Web Merchant Application Service]
2. Need a developer account for App

1. Log in to your Apple Developer account. If you don't have one, create one now. Click "Certificates, Identifiers & Profiles."

2. Select identifiers and click + button.
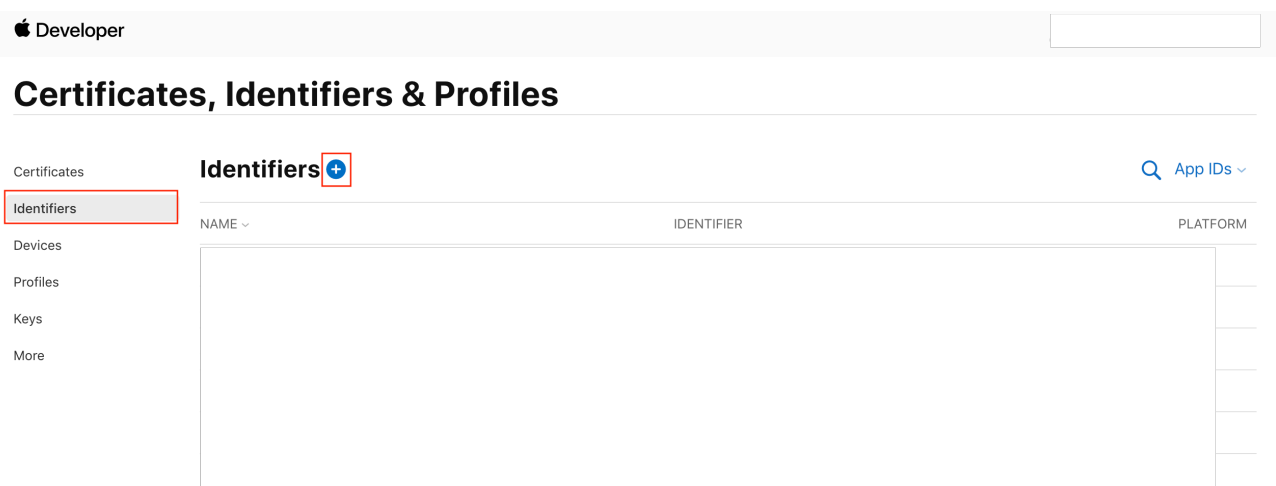
3. Select Merchant IDs click Continue button.

‹ All Identifiers

Register a New Identifier   [Continue]

○ **App IDs**
Register an App ID to enable your app to access available services and identify your app in a provisioning profile. You can enable app services when you create an App ID or modify these settings later.

○ **Services IDs**
For each website that uses Sign In with Apple, register a services identifier (Services ID), configure your domain and return URL, and create an associated private key.

○ **Pass Type IDs**
Register a pass type identifier (Pass Type ID) for each kind of pass you create (i.e. gift cards). Registering your Pass Type IDs lets you generate Apple-issued certificates which are used to digitally sign and send updates to your passes, and allow your passes to be recognized by Wallet.

○ **Website Push IDs**
Register a Website Push Identifier (Website Push ID). Registering your Website Push IDs lets you generate Apple-issued certificates which are used to digitally sign and send push notifications from your website to macOS.

○ **iCloud Containers**
Registering your iCloud Container lets you use the iCloud Storage APIs to enable your apps to store data and documents in iCloud, keeping your apps up to date automatically.

○ **App Groups**
Registering your App Group allows access to group containers that are shared among multiple related apps, and allows certain additional interprocess communication between the apps.

◉ **Merchant IDs**
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 Payment Processing certificate for each registered Merchant ID to validate transactions initiated within your app.

4. Fill out the appropriate information, click Continue button.

Certificates, Identifiers & Profiles

‹ All Identifiers

Register a Merchant ID   [Back] [Continue]

Description   Identifier

| TapPay docs test | | merchant.docs.test |

You cannot use special characters such as @, &, *, ', "   We recommend using a reverse-domain name style string (i.e., com.domainname.appname).

5. Double check your Merchant ID information, click Register button, create Merchant ID success.

Certificates, Identifiers & Profiles

‹ All Identifiers

Register a Merchant ID   [Back] [Register]

Description   Identifier
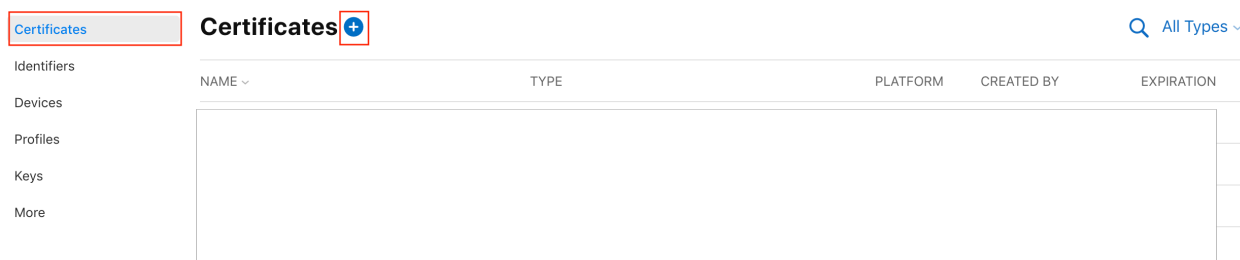TapPay docs test   merchant.docs.test

Create Payment Processing Certificate

1. Select Certificate click + button.

Certificates, Identifiers & Profiles

Certificates	**Certificates** ➕			Q All Types ⌄
Identifiers				
	NAME ⌄	TYPE	PLATFORM	CREATED BY	EXPIRATION
Devices				
Profiles				
Keys				
More				

2. Select Service > Apple Pay Payment Processing Certificate, click Continue button.

   ### Create a New Certificate   `Continue`

   ○ **macOS Apple Push Notification service SSL (Production)**
   Establish connectivity between your notification server and the Apple Push Notification service production environment. A separate certificate is required for each app you distribute.

   ○ **Pass Type ID Certificate**
   Sign and send updates to passes in Wallet.

   ○ **Website Push ID Certificate**
   Sign and send updates for Websites.

   ○ **WatchKit Services Certificate**
   Establish connectivity between your notification server, the Apple Push Notification service sandbox, and production environment to update ClockKit complication data. When utilizing HTTP/2, the same certificate can be used to deliver app notifications, update ClockKit complication data, and alert background VoIP apps of incoming activity. A separate certificate is required for each app you distribute.

   ○ **VoIP Services Certificate**
   Establish connectivity between your notification server, the Apple Push Notification service sandbox, and production environment to alert background VoIP apps of incoming activity. A separate certificate is required for each app you distribute.

   ● **Apple Pay Payment Processing Certificate**
   Decrypt app transaction data sent by Apple to a merchant/developer.

   ○ **Apple Pay Merchant Identity Certificate**
   A client TLS certificate that is used to authenticate you to Apple Pay Payment Processing Servers

3. Select the desired Merchant ID, click Continue button.

   ## Certificates, Identifiers & Profiles

   ‹ All Certificates

   ### Create a New Certificate   `Back` `Continue`

   Select a Merchant ID for your Apple Pay Payment Processing Certificate
   Each Merchant ID you identify within your app will require its own individual Apple Pay Payment Processing Certificate. The certificate is used by Apple to encrypt transaction data sent by Apple back to the merchant.

   Merchant ID:
T9G64ZZGC4.merchant.docs.test	⌄

16

4. Click Apple Pay Payment Processing Certificate > Create Certificate button.

‹ All Identifiers

Edit or Configure Merchant ID   Remove   Save

Name   Identifier
TapPay docs test   merchant.docs.test
You cannot use special characters such as @, &, *, ', "

Apple Pay Payment Processing Certificate

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

Create an Apple Pay Payment Processing Certificate for this Merchant ID.

Create Certificate

Apple Pay Payment Processing on the Web

To configure Apple Pay Payment Processing on the web for this merchant ID, you must register and verify the domains that will process transactions. You must also create a Apple Pay Merchant Identity, which authenticates your web sessions with the Apple Pay Payment Processing servers.

Incorporation of Apple Pay Payment Processing into your website is subject to these Apple Pay Payment Processing Web Merchant Terms and Conditions and Acceptable Use Guidelines. Failure to comply with any of these Terms and Conditions or guidelines may result in deactivation of Apple Pay Payment Processing transactions on your website.

5. Click Continue button.
   Please choose No. TapPay doesn't support Union Pay for Apple Pay so far.

‹ All Identifiers

Edit or Configure Merchant ID   Remove   Continue

Name   Identifier
TapPay docs test   merchant.docs.test
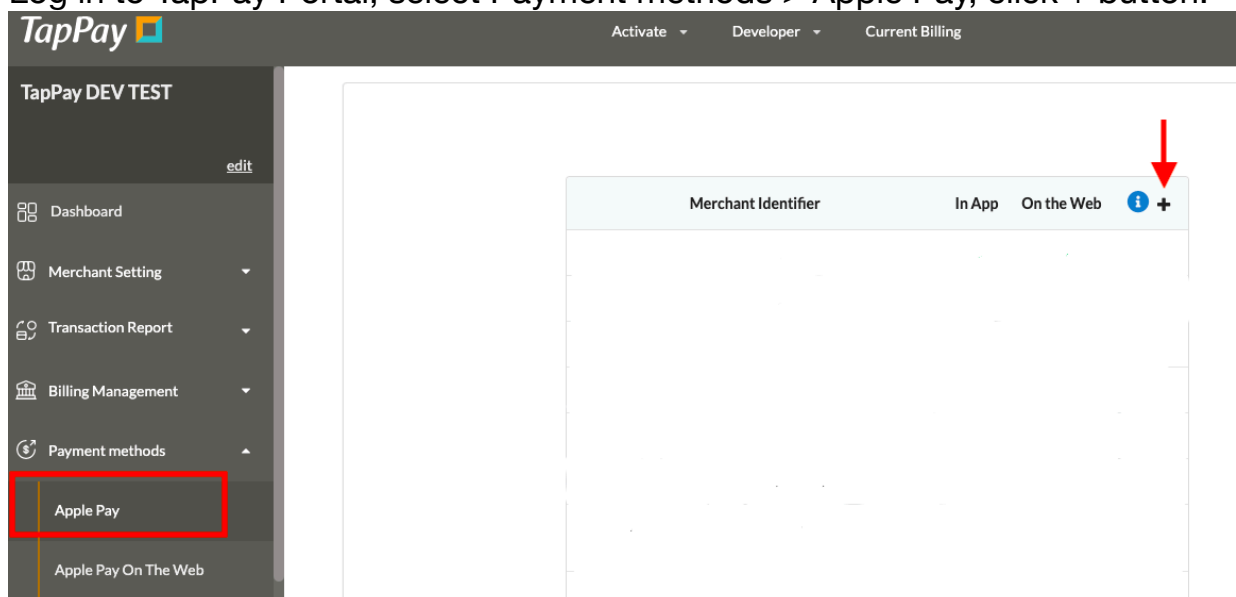
Will payments associated with this Merchant ID be processed exclusively in China?

● No
○ Yes

6. Log in to TapPay Portal, select Payment methods > Apple Pay, click + button.

7. Click payment_processing_certificate.csr download file, Click next button.

Create Apple Pay Merchant

1 Download CSR   **2** Upload Certificate

To create a Payment Processing Certificate for Apple Pay, download the CSR file and upload to Apple Developer

⬇ payment_processing_certificate.csr

Next

8. See the following screen, continue to step 9

Create Apple Pay Merchant

✓ Download CSR   **2** Upload Certificate

1. Go to Apple Developer
2. Create a Merchant ID if you don't have one
3. Edit Merchant ID
4. Create Payment Processing Certificate
5. Upload the CSR file you just download
   (payment_processing_certificate.csr)
6. Download Payment Processing Certificate (apple_pay.cer)
7. Upload apple_pay.cer using the button on the right
8. Visit our Document for more information

📄 Select apple_pay.cer

⬆ Upload

9. Go back to apple pay developer, upload your CSR file by clicking "Choose File…" Click "Continue".

‹ All Certificates

Create a New Certificate   Back   Continue

Certificate Name
Apple Pay Payment Processing Certificate

Upload a Certificate Signing Request
To manually generate a Certificate, you need a Certificate Signing Request (CSR) file from your Mac.
Learn more ›

Choose File   payment_processing_certificate (5).csr

10. Click download button download apple_pay.cer file.

11. Back to TapPay Portal, click choose apple_pay.csr button and upload the resulting .cer file.

12. On the Apple Pay page, you can see the newly created Apple Pay in App, and you can start to develop (refer to Frontend> Apple Pay in Apps); if you need to create Apple Pay on the Web, please complete the follow-up actions

Create Merchant Identifier Certificate (Required only if you apply for developer account in person)

1. Select Certificate click + button.

Certificates, Identifiers & Profiles

Certificates	Certificates ⊕				All Types ∨
Identifiers	NAME ∨	TYPE	PLATFORM	CREATED BY	EXPIRATION
Devices					
Profiles					
Keys					
More					

2. Select Apple Pay Merchant Identity Certificate, Click Continue button.

Create a New Certificate   Continue

Services

○ **iOS Apple Push Notification service SSL (Sandbox)**
Establish connectivity between your notification server and the Apple Push Notification service sandbox environment to deliver remote notifications to your app. A separate certificate is required for each app you develop.

○ **macOS Apple Push Notification service SSL (Sandbox)**
Establish connectivity between your notification server and the Apple Push Notification service sandbox environment. A separate certificate is required for each app you develop.

○ **Apple Push Notification service SSL (Sandbox & Production)**
Establish connectivity between your notification server, the Apple Push Notification service sandbox, and production environments to deliver remote notifications to your app. When utilizing HTTP/2, the same certificate can be used to deliver app notifications, update ClockKit complication data, and alert background VoIP apps of incoming activity. A separate certificate is required for each app you distribute.

○ **macOS Apple Push Notification service SSL (Production)**
Establish connectivity between your notification server and the Apple Push Notification service production environment. A separate certificate is required for each app you distribute.

○ **Pass Type ID Certificate**
Sign and send updates to passes in Wallet.

○ **Website Push ID Certificate**
Sign and send updates for Websites.

○ **WatchKit Services Certificate**
Establish connectivity between your notification server, the Apple Push Notification service sandbox, and production environment to update ClockKit complication data. When utilizing HTTP/2, the same certificate can be used to deliver app notifications, update ClockKit complication data, and alert background VoIP apps of incoming activity. A separate certificate is required for each app you distribute.

○ **VoIP Services Certificate**
Establish connectivity between your notification server, the Apple Push Notification service sandbox, and production environment to alert background VoIP apps of incoming activity. A separate certificate is required for each app you distribute.

○ **Apple Pay Payment Processing Certificate**
Decrypt app transaction data sent by Apple to a merchant/developer.

◉ **Apple Pay Merchant Identity Certificate**
A client TLS certificate that is used to authenticate you to Apple Pay Payment Processing Servers

3. Select desired MerchantID, click Continue button.

Create a New Certificate   Back   Continue

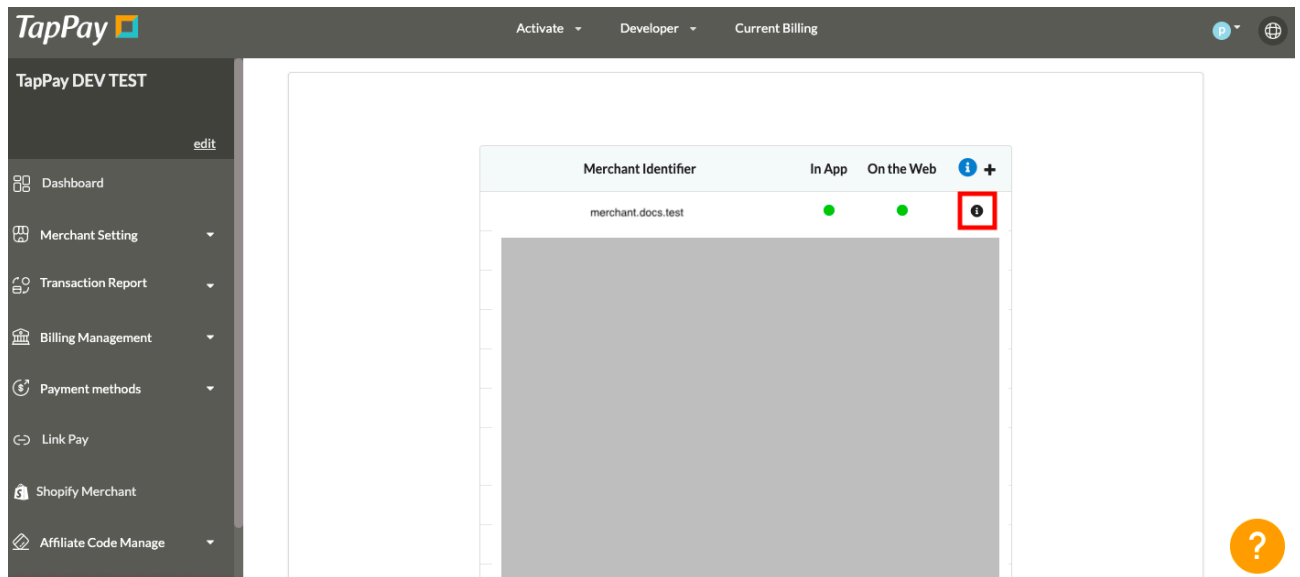**Select a Merchant ID for your Apple Pay Merchant Identity Certificate**
Each Merchant ID you identify within your app will require its own individual Apple Pay Merchant Identity Certificate. The certificate is used is used to authenticate you to Apple Pay Payment Processing servers.

Merchant ID:

T9G64ZZGC4.merchant.docs.test

4. Log in to TapPay Portal Apple Pay, select desired Apple merchant identifier and click i.

5. Click Apple Pay on the web > Upload Certificate

6. Download merchant_identity_certificate.csr

7. See the following screen, continue to step 8

8. Upload your CSR file by clicking "Choose File…"

9. Go back to TapPay Portal, Click Download.

‹ All Certificates

Download Your Certificate   [Revoke] [Download]

Certificate Details

Certificate Name   Certificate Type   Download your certificate to your Mac, then double click the .cer file to install in
merchant.docs.test   Apple Pay Merchant Identity   Keychain Access. Make sure to save a backup copy of your private and public
                                                                        keys somewhere secure.
Expiration Date   Created By
2021/07/20

10. Upload the resulting .cer file back to TapPay Portal.

Update Merchant Identity Certificate

✔ Download CSR   **2** Upload Certificate

1. Go to Apple Developer   **1.** [Select merchant_id.cer]
2. Edit Merchant ID: **merchant.docs.test**
3. Create Merchant Identity Certificate
4. Upload the CSR file you just download
   (payment_processing_certificate.csr)
5. Download Merchant Identity Certificate (merchant_id.cer)
6. Upload merchant_id.cer using the button on the right
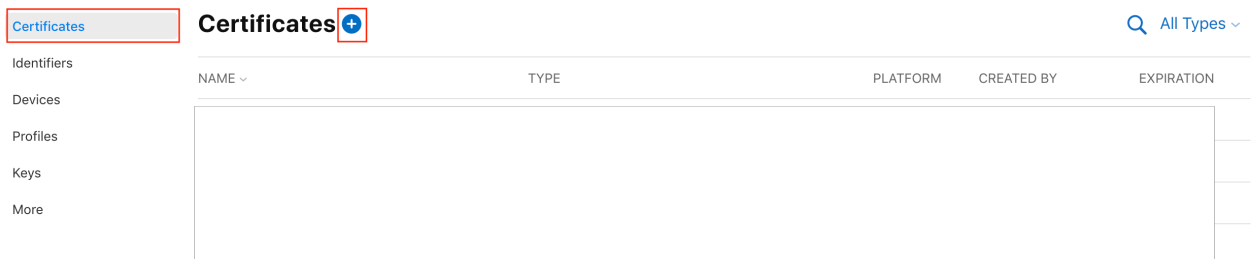7. Visit our Document for more information

2. [⬆ Upload]

Add Domain (Required only if you apply for developer account in person)

1. Select Certificate click + button.

Certificates, Identifiers & Profiles

2. Select Service > Apple Pay Payment Processing Certificate, click Continue button.

3. Select desired MerchantID, Click Continue button.

Certificates, Identifiers & Profiles

4. Click Add Domain button.

Apple Pay Payment Processing on the Web

To configure Apple Pay Payment Processing on the web for this merchant ID, you must register and verify the domains that will process transactions. You must also create a Apple Pay Merchant Identity, which authenticates your web sessions with the Apple Pay Payment Processing servers.

Incorporation of Apple Pay Payment Processing into your website is subject to these Apple Pay Payment Processing Web Merchant Terms and Conditions and Acceptable Use Guidelines. Failure to comply with any of these Terms and Conditions or guidelines may result in deactivation of Apple Pay Payment Processing transactions on your website.

Merchant Domains

Add a domain for use with this Merchant ID.

[Add Domain]

Apple Pay Merchant Identity Certificate

Name: merchant.docs.test
Type: Apple Pay Merchant Identity
Expires: Jul 19, 2021

[Revoke]   [Download]

Create an Apple Pay Merchant Identity Certificate for this Merchant ID.

[Create Certificate]

5. Fill out your domain.

‹ View Identifier

Register   [Save]

To use Apple Pay Payment Processing on the web you must register and verify your fully qualified domain name. To begin enter your domain below.

Enter the domain you wish to register

https:// | tappaysdk.docs |

6. Download the verification file and host it in the designated path. Click "Verify" after you are done.

‹ View Identifier

**Verify**   [Download] [Verify]

Instructions and Information
To verify ownership of your domain download the file and place it at the location listed below. After uploading your file continue with the verification step.

We'll look for your file at the following location:

https://tappaysdk.docs/.well-known/apple-developer-merchantid-domain-association.txt

25

7. Go back to TapPay Portal payment methods> Apple Pay, select the Apple Pay merchant ID you want to use, and click i
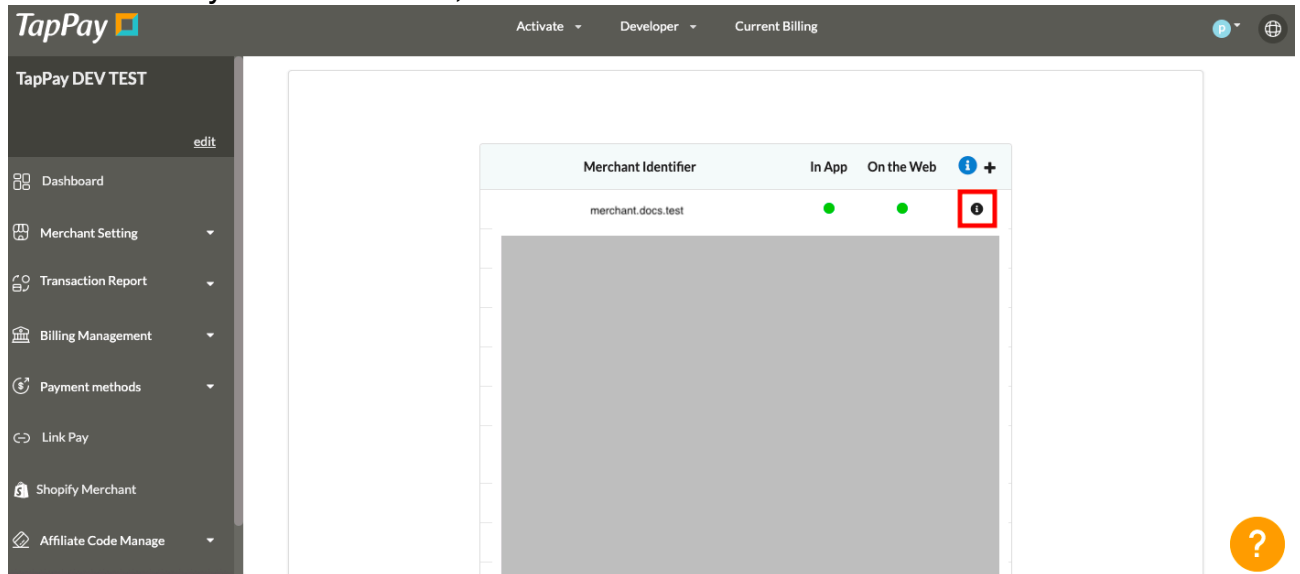
8. Go to Apple Pay on the web, fill out the domain, click save

9. Apple Pay on the web is established. Please go to Frontend> Apple Pay on the Web to start develop
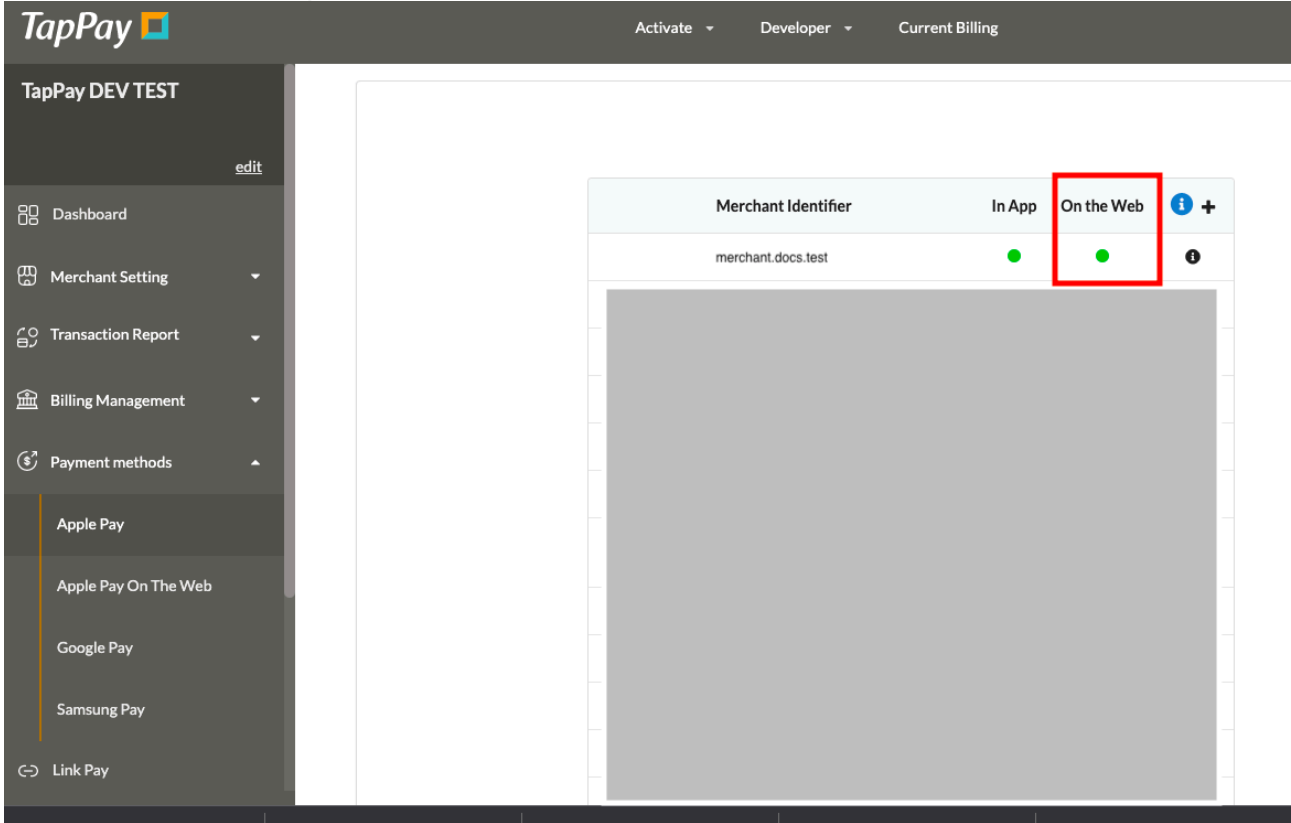
Apple Pay - QA

Q1 : How to solve the following error when create Merchant ID ?

This error means that your naming does not comply with Apple regulations. Please change your name and press Register
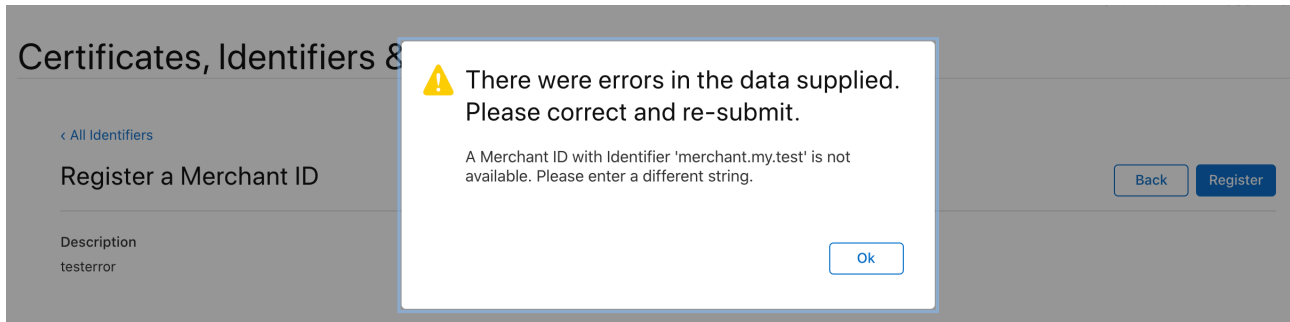
Q2 : How to renew the certificate ?

Apple will notify the merchant one month before the certificate expires, you can update the certificate according to the following process

1. Go to TapPay Portal, select the Apple Merchant ID of the voucher to be updated, and click i

2. Click Update Certificate, select the certificate just created, upload the update certificate

Merchant Identifier:

Apple Pay

Payment Processing Certificate

Name: [   ]

Type: Apple Pay Payment Processing Certificate

Expires: [   ]

⬆ Update Certificate

Apple Pay on the Web

Domain

test.com.tw

Merchant Identity Certificate

Name: [   ]

Type: Merchant Identity Certificate

Expires: [   ]

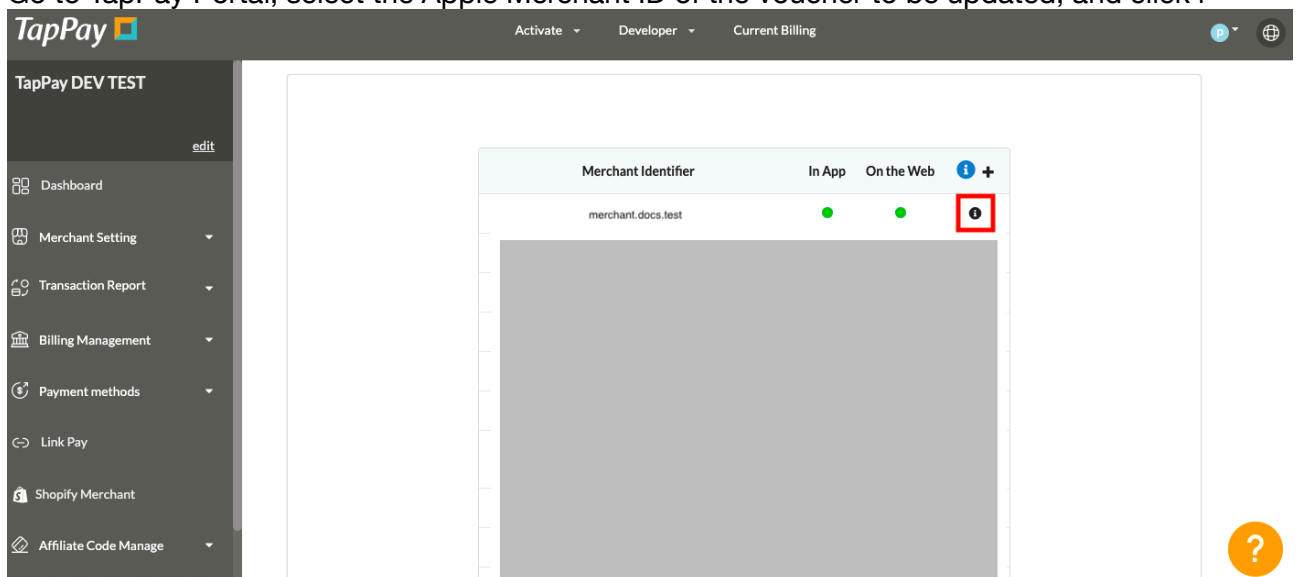⬆ Upload Certificate

↩ Back   🗑 Delete   💾 Save

3. Please re-apply for Payment Processing Certificate and Merchant Identify Certificate

   *When you apply for a new certificate, the following screen will appear
   Please click Activate （The old certificate will be revoke）

‹ All Certificates

Download Your Certificate   Activate   Revoke   Download

Certificate Details

Certificate Name   Certificate Type   Download your certificate to your Mac, then double click the .cer file to install in
[   ]   Apple Pay Payment Processing   Keychain Access. Make sure to save a backup copy of your private and public
                                                                 keys somewhere secure.
Expiration Date   Created By
2022/06/21   [   ]

(When you already have two sets of certificates, Apple will ask you if you want to activate the new certificate)

Certificates, Identifiers &

‹ All Certificates

Download Your Certificate

Activate Certificate

Activating this certificate will result in the revocation of all previous certificates generated for this Merchant Identifier. Certificate revocation will cause transaction failure within your app. Ensure this certificate is configured properly on your servers prior to activating to prevent transaction processing failure.

Cancel   Activate

Activate   Revoke   Download

Certificate Details

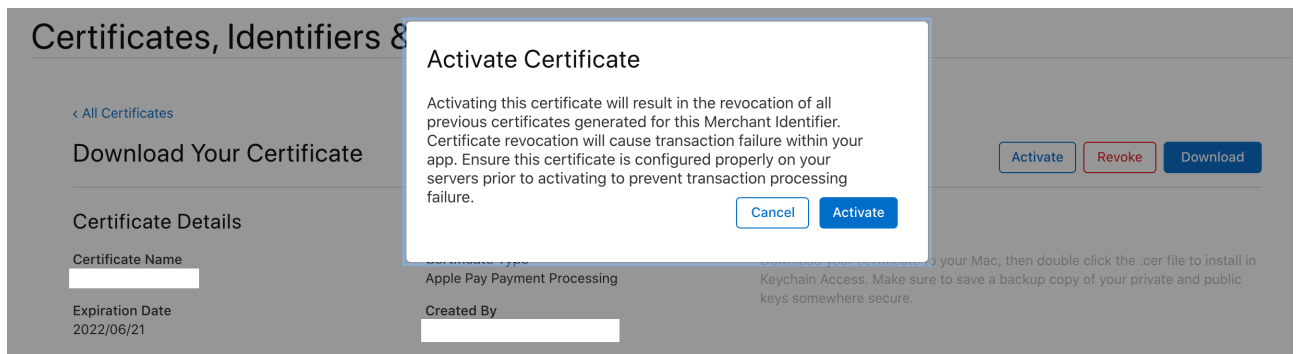Certificate Name

Expiration Date
2022/06/21

Certificate Type
Apple Pay Payment Processing

Created By

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.

Q3 : Domain verification failed

If continuous verification is not possible, please cooperate with your IT staff to perform the following tests

1. Use SSL checker (https://www.sslshopper.com/ssl-checker.html) ，Confirm whether the SSL certificate has reached A + level
2. All pages containing Apple Pay must have HTTPS, which is necessary in both formal and test environments. In the development environment, you can meet the needs of HTTPS through the ngrok service. You can refer to our Ngrok tutorial.