

# Setting up Apple Pay Merchant Certificates 2011 R2.0

When you register for Apple Pay you will receive a certificate signing request (.csr file) from Worldpay. This is linked to the merchant id

1. Register Merchant ID at <https://developer.apple.com/account/resources/identifiers/add/merchant>
  - a. The description field is for internal use. Hint: use sandbox and production so you can make distinction between test and production Apple Pay merchants
  - b. The identifier field should be the Merchant ID you've received from Worldpay
2. Create the certificates for the merchant at <https://developer.apple.com/account/resources/certificates/list>
  - a. Select 'development' from the All Types menu if you are registering your certificates for the sandbox
  - b. Click the + to register certificates
  - c. Create a 'Apple Pay Merchant Identity Certificate'
  - d. Select your Merchant ID
  - e. Upload the Certificate Signing Request you've received from Worldpay
  - f. Download the certificate file. Keep this somewhere safe
3. Domain validation:
  - a. You will need to host a text file on your test / production environment. This needs to be open to public or at least to Apple - [https://developer.apple.com/documentation/apple\\_pay\\_on\\_the\\_web/setting\\_up\\_your\\_server](https://developer.apple.com/documentation/apple_pay_on_the_web/setting_up_your_server)
  - b. Edit your Merchant ID
  - c. Add your frontend domain(s) and you will receive a text file.
  - d. Add this to your webserver at the right location (written on the page)
  - e. Note that this has to be with a valid SSL and return text/plain
  - f. Press the verify button
4. Create certificates
  - a. Create a public and private key - `openssl req -out apple.csr -new -newkey rsa:2048 -nodes -keyout sandbox.key`
  - b. Upload the file `apple.csr` to the Apple Pay Merchant Identity Certificate
  - c. Download the `merchant_id.cer` file you receive from Apple
  - d. Create a pem file based on this certificate `openssl x509 -inform der -in merchant_id.cer -out sandbox.pem`
  - e. Create a p12 file to be used in the Java backend `openssl pkcs12 -export -in sandbox.pem -inkey sandbox.key -out apple-pay.p12 -name "Worldpay <your name> Apple Pay keystore"`
    - i. Secure your p12 file with a password!
5. Save the `apple-pay.p12` in `ext-worldpay/worldpayapi/resources/applepaycert`
6. Update your `worldpayApplePayHttpClient` to have the right password and p12 file.