

# PayFast Developer Documentation>

**Live:** <https://www.payfast.co.za/eng/process>

**Sandbox:** <https://sandbox.payfast.co.za/eng/process>

## Simple Form Integration

```
<form action="https://www.payfast.co.za/eng/process" method="post">
  <input type="hidden" name="merchant_id" value="10000100">
  <input type="hidden" name="merchant_key" value="46f0cd694581a">
  <input type="hidden" name="amount" value="100.00">
  <input type="hidden" name="item_name" value="Test Product">
  <input type="submit">
</form>
```

## Step 4: Confirm payment is successful

Before the customer is returned to the URL you specified in the “return\_url” parameter, Payfast will send a notification to your “notify\_url” page.

### 4.1. Setting up your notification page:

On receiving the payment notification from Payfast, return a header 200 to prevent further retries.

If no 200 response is returned the notification will be re-sent immediately, then after 10 minutes and then at exponentially longer intervals until eventually stopping.

### 4.2. Receive the notification parameters

As part of the notification you can expect to receive the following parameters:

#### Transaction details

m\_payment\_idstring, 100 charOPTIONAL

Unique payment ID on the merchant’s system.

pf\_payment\_idintegerREQUIRED

Unique transaction ID on Payfast.

payment\_statusstring (CANCELLED / COMPLETE)REQUIRED

After a successful payment the status sent will be COMPLETE. When a subscription is cancelled the status will be CANCELLED.

item\_namestring, 100 charREQUIRED

The name of the item being charged for, or in the case of multiple items the order number.

item\_descriptionstring, 255 charOPTIONAL

The description of the item being charged for, or in the case of multiple items the order description.

amount\_grossdecimalOPTIONAL

The total amount the customer paid in ZAR.

amount\_feedecimalOPTIONAL

The total in fees which was deducted from the amount in ZAR.

amount\_netdecimalOPTIONAL

The net amount credited to the merchant's account in ZAR.

custom\_int<1..5>integer, 255 charOPTIONAL

The series of 5 custom integer variables (custom\_int1, custom\_int2...) originally passed by the receiver during the payment request.

custom\_str<1..5>string, 255 charOPTIONAL

The series of 5 custom string variables (custom\_str1, custom\_str2...) originally passed by the receiver during the payment request.

## Customer details

name\_firststring, 100 charOPTIONAL

The customer's first name.

name\_laststring, 100 charOPTIONAL

The customer's last name.

email\_addressstring, 100 charOPTIONAL

The customer's email address.

## Merchant details

merchant\_idinteger, 8 charREQUIRED

The Merchant ID as given by the Payfast system. Used to uniquely identify the receiver's account.

## Recurring Billing details

tokenstring, 36 charREQUIRED

Unique ID on Payfast that represents the subscription.

billing\_datedate, YYYY-MM-DDOPTIONAL

The date from which future subscription payments will be made. Eg. 2020-01-01. Defaults to current date if not set.

## Security information

signatureMD5 hash in lower caseOPTIONAL

A security signature of the transmitted data taking the form of an MD5 hash of all the url encoded submitted variables.

### Example of a Transaction Webhook Payload

Select library

```
$ITN_Payload = [  
  'm_payment_id' => 'SuperUnique1',  
  'pf_payment_id' => '1089250',  
  'payment_status' => 'COMPLETE',  
  'item_name' => 'test+product',  
  'item_description' => ,  
  'amount_gross' => 200.00,  
  'amount_fee' => -4.60,  
  'amount_net' => 195.40,  
  'custom_str1' => ,  
  'custom_str2' => ,  
  'custom_str3' => ,  
  'custom_str4' => ,  
  'custom_str5' => ,  
  'custom_int1' => ,  
  'custom_int2' => ,
```

```
'custom_int3' => ,
'custom_int4' => ,
'custom_int5' => ,
'name_first' => ,
'name_last' => ,
'email_address' => ,
'merchant_id' => '10012577',
'signature' => 'ad8e7685c9522c24365d7ccea8cb3db7'
];
```

### 4.3. Conduct security checks

Conduct four security checks to ensure that the data you are receiving is correct, from the correct source and hasn't been altered; you should not continue the process if a test fails!

#### Notification page

Select library

```
<?php
// Tell Payfast that this page is reachable by triggering a header 200
header( 'HTTP/1.0 200 OK' );
flush();

define( 'SANDBOX_MODE', true );
$pfHost = SANDBOX_MODE ? 'sandbox.payfast.co.za' : 'www.payfast.co.za';
// Posted variables from ITN
$pfData = $_POST;

// Strip any slashes in data
foreach( $pfData as $key => $val ) {
    $pfData[$key] = stripslashes( $val );
}

// Convert posted variables to a string
foreach( $pfData as $key => $val ) {
    if( $key !== 'signature' ) {
        $pfParamString .= $key .'=' . urlencode( $val ) . '&';
    } else {
        break;
    }
}

$pfParamString = substr( $pfParamString, 0, -1 );
```

- **Verify the signature**

Verify the security signature in the notification; this is done in a similar way that the signature that you generated for stage one of the user payment flow.

The string that gets created needs to include all fields posted from Payfast.

#### Verify Signature

Select library

```
<?php
function pfValidSignature( $pfData, $pfParamString, $pfPassphrase = null ) {
    // Calculate security signature
    if($pfPassphrase === null) {
        $tempParamString = $pfParamString;
    } else {
        $tempParamString = $pfParamString.'&passphrase='.urlencode( $pfPassphrase );
    }

    $signature = md5( $tempParamString );
    return ( $pfData['signature'] === $signature );
}
```

- **Check that the notification has come from a valid Payfast domain**

The following is a list of valid domains:

- www.payfast.co.za
- w1w.payfast.co.za
- w2w.payfast.co.za
- sandbox.payfast.co.za

#### Check valid Payfast domain

Select library

```
<?php
function pfValidIP() {
    // Variable initialization
    $validHosts = array(
        'www.payfast.co.za',
        'sandbox.payfast.co.za',
        'w1w.payfast.co.za',
        'w2w.payfast.co.za',
    );

    $validIps = [];

    foreach( $validHosts as $pfHostname ) {
        $ips = gethostbyname( $pfHostname );

        if( $ips !== false )
            $validIps = array_merge( $validIps, $ips );
    }

    // Remove duplicates
    $validIps = array_unique( $validIps );
    $referrerIp = gethostbyname(parse_url($_SERVER['HTTP_REFERER'])['host']);
    if( in_array( $referrerIp, $validIps, true ) ) {
        return true;
    }
    return false;
}
```

- **Compare payment data**

The amount you expected the customer to pay should match the “amount\_gross” value sent in the notification.

#### Compare payment data

Select library

```
<?php
function pfValidPaymentData( $cartTotal, $pfData ) {
    return !(abs((float)$cartTotal - (float)$pfData['amount_gross']) > 0.01);
}
```

- **Perform a server request to confirm the details**

Validate the data that you have received from Payfast by contacting our server and confirming the order details.

**Live:** <https://www.payfast.co.za/eng/query/validate>

**Sandbox:** <https://sandbox.payfast.co.za/eng/query/validate>

#### Curl Options

CURLOPT_RETURNTRANSFER	Set to TRUE to return the transfer as a string of the return value of curl_exec() instead of outputting it out directly.
CURLOPT_HEADER	Set to FALSE to exclude the header in the output.
CURLOPT_SSL_VERIFYHOST	Set to 2 to check the existence of a common name in the SSL peer certificate, and to also verify that it matches the hostname provided.

CURLOPT_SSL_VERIFYPEER	This option determines whether curl verifies the authenticity of the peer's certificate. The default value is 1.
CURLOPT_URL	Set to the Payfast query validation URL, eg. <a href="https://www.payfast.co.za/eng/query/validate">https://www.payfast.co.za/eng/query/validate</a> . This can also be set when initializing a session with <code>curl_init()</code> .
CURLOPT_POST	Set to TRUE to do a regular HTTP POST. This POST is the normal application/x-www-form-urlencoded kind, most commonly used by HTML forms.
CURLOPT_POSTFIELDS	The full data to post via HTTP to Payfast. This parameter is passed as a urlencoded string, such as: <code>m_payment_id=01&amp;pf_payment_id=01234..</code>

**Server request to confirm details**

## Select library

```
<?php
function pfValidServerConfirmation( $pfParamString, $pfHost = 'sandbox.payfast.co.za', $pfProxy = null ) {
    // Use cURL (if available)
    if( in_array( 'curl', get_loaded_extensions(), true ) ) {
        // Variable initialization
        $url = 'https://'. $pfHost .'/eng/query/validate';

        // Create default cURL object
        $ch = curl_init();

        // Set cURL options - Use curl_setopt for greater PHP compatibility
        // Base settings
        curl_setopt( $ch, CURLOPT_USERAGENT, NULL ); // Set user agent
        curl_setopt( $ch, CURLOPT_RETURNTRANSFER, true ); // Return output as string rather than outputting it
        curl_setopt( $ch, CURLOPT_HEADER, false ); // Don't include header in output
        curl_setopt( $ch, CURLOPT_SSL_VERIFYHOST, 2 );
        curl_setopt( $ch, CURLOPT_SSL_VERIFYPEER, true );

        // Standard settings
        curl_setopt( $ch, CURLOPT_URL, $url );
        curl_setopt( $ch, CURLOPT_POST, true );
        curl_setopt( $ch, CURLOPT_POSTFIELDS, $pfParamString );
        if( !empty( $pfProxy ) )
            curl_setopt( $ch, CURLOPT_PROXY, $pfProxy );

        // Execute cURL
        $response = curl_exec( $ch );
        curl_close( $ch );
        if ( $response === 'VALID' ) {
            return true;
        }
    }
    return false;
}
```

**Bringing the checks together**

## Select library

```
<?php
$check1 = pfValidSignature($pfData, $pfParamString);
$check2 = pfValidIP();
$check3 = pfValidPaymentData($cartTotal, $pfData);
$check4 = pfValidServerConfirmation($pfParamString, $pfHost);

if($check1 && $check2 && $check3 && $check4) {
    // All checks have passed, the payment is successful
} else {
    // Some checks have failed, check payment manually and log for investigation
}
```