

CONTENTS

VERSION HISTORY	4
INTRODUCTION.....	4
BENEFITS TO PAYWEB	4
PAYGATE ACCOUNT SETUP OPTIONS – PER PAYGATEID	5
PASSWORD & CARD TYPES ACCEPTED	5
AUTO-SETTLE: DEFAULT IS ON	5
PAYPROTECTOR: DEFAULT IS NOT ACTIVATED	5
PAYMENT CONFIRMATION: DEFAULT IS ACTIVATED WITH NO BCC	5
PAYVAULT: DEFAULT IS NOT ACTIVATED	6
SETUP OPTIONS WHEN MORE THAN ONE PAYMENT METHOD IS ACTIVATED	6
PROCESS FLOW DIAGRAM	7
PROCESS FLOW DESCRIPTION	7
THE LANDING PAGES	8
PAYMENT MENU PAGE	8
PAYMENT PAGE	8
REQUEST AND RESPONSE DATA	9
STEP 1 – TRANSACTION REQUEST POSTED TO PAYWEB (THE REQUEST)	10
STEP 2 – RESPONSE TO STEP 1 WITH PAY_REQUEST_ID	12
STEP 3 – RE-DIRECT CLIENT TO PAYWEB WITH PAY_REQUEST_ID	13
MISCELLANEOUS INFORMATION.....	18
SECURITY	18
CHECKSUM EXAMPLES	19
<i>Step 1 - Checksum Examples.....</i>	<i>19</i>
<i>Step 2 - Checksum Example.....</i>	<i>19</i>
<i>Step 3 - Checksum Example.....</i>	<i>20</i>
<i>Step 4 - Checksum Examples.....</i>	<i>20</i>
<i>Step 5 – No Checksum</i>	<i>20</i>
<i>Step 6 - Checksum Example.....</i>	<i>20</i>
<i>Step 7 - Checksum Example.....</i>	<i>21</i>
REQUEST AND RESPONSE EXAMPLES	22
<i>Step 1 - Request Examples.....</i>	<i>22</i>
<i>Step 4 - Response Example.....</i>	<i>23</i>
TESTING.....	25
MASTERCARD SECURECODE & VERIFIED BY VISA.....	26
<i>A typical credit card authorisation flow including PayProtector and 3D.....</i>	<i>26</i>
FREQUENTLY ASKED QUESTIONS	28

HOW DO I KNOW THE TRANSACTION IS APPROVED?	28
CAN I DO THE AUTHORISATION AND THE SETTLEMENT SEPARATELY?	28
WHAT RESPONSE IS RETURNED IF THE CUSTOMER CLICKS THE 'CANCEL' BUTTON ON THE PAYWEB PAYMENT PAGE?	28
HOW WILL I KNOW THAT I THE TRANSACTION WAS AUTHENTICATED AND I HAVE CHARGE BACK PROTECTION?	28
THE TRANSACTION WAS AUTHENTICATED AND DECLINED; HOW CAN THIS BE?	28
IS IT POSSIBLE TO NOT USE VERIFIED-BY-VISA AND MASTERCARD SECURECODE?	28
PAYVAULT VALIDATION INFORMATION	29
APPENDIX A : CODES & DESCRIPTIONS	30
RESULT CODES	30
TRANSACTION STATUS.....	32
MASTERCARD SECURECODE / VERIFIED BY VISA AUTHENTICATION INDICATOR	32
PAYMENT METHOD CODES	32
LOCALE CODES	32
COUNTRY AND CURRENCY CODES	34

Version History

Version	Date	Comment
1.00	August 2012	New release of PayWeb.
1.01	August 2012	Updated flow diagram to show 'OK' status returned by merchant web site (step 5).
1.02	July 2013	Added M-Pesa to Testing page.
1.03	November 2013	Updated requests and responses to allow for PayVault tokenisation and processing of payments with tokens.
1.03.1	March 2016	Updated test account information.
1.03.2	March 2016	Update images, examples and format
1.03.3	April 2016	Update to URL's and response data in examples
1.03.4	March 2017	Overall Update to request fields, Notify URL PORT note and general grammar corrections.

Introduction

PayWeb is a secure payment system hosted by PayGate. A single integration to PayWeb gives you access to multiple payment methods. PayGate is a PCI compliant payment service provider.

PayGate is continually adding to the list of available payment methods. Please contact the PayGate Support team (email: support@paygate.co.za) to confirm which payment methods are available in your locale.

Benefits of PayWeb

- The setup process is relatively simple and can be easily integrated into existing web sites.
- PayWeb can be customized to suit the look and feel of your website.
- Multiple payment methods are accessible via a single PayWeb integration.
- Multiple payment options can be offered to the client via a menu system or PayGate can provide the merchant with many PayGateID's so that the merchant can display the choice of payment methods to the client.
- MasterCard SecureCode and Verified-by-Visa cardholder authentication built in to minimize the risk of charge backs for credit card transactions.
- PayGate's PayProtector fraud and risk system is supported for real-time fraud screening and reporting.
- Using PayWeb will reduce a merchant's PCI compliance scope as all sensitive data is captured in the PayGate environment. PayGate is certified PCI DSS Level 1 compliant.
- PayVault credit card tokenisation can be activated on PayWeb, and PayWeb allows for PayVault tokens to be submitted in the PayWeb request to allow for payments without the card holder having to enter their card number or expiry date (a CVV will be required).

PayGate account setup options – per PayGateID

The following parameters can be configured for each PayGate account (i.e. per PayGateID). These are agreed and pre-set during the application process and are configured when our Support team sets up your PayGate account, or can be configured via the Back-Office merchant administration website once the account is live.

Password & Card Types Accepted

Merchants are given access to the PayWeb configuration page (via the PayGate BackOffice) where they set the following options:

- The Encryption Key used in the checksum calculation.
- Choose which credit card brands to accept. MasterCard and Visa are enabled by default if the credit card payment method is activated.

Auto-Settle: Default is ON

Applies to: Card processing

With this option enabled, you do not need to send a Settlement transaction for an approved Authorisation. As soon as the bank approves the Authorisation, PayGate immediately and automatically creates the Settlement transaction on your behalf. This option is enabled by default.

PayProtector: Default is Not Activated

Applies to: Card processing.

PayProtector is PayGate's fraud and risk system, designed to help the merchant minimize the risk of loss from fraudulent transactions. Fraud has become a serious problem and often adds significant costs for internet merchants. PayProtector scrutinizes transactions from several angles combining internal, local and international information to identify, report on, and / or block fraudulent transactions.

Payment Confirmation: Default is Activated with no Bcc

Applies to: All transaction processing.

By default, PayGate will send a Payment Confirmation email to the customer's email address for each approved transaction. If this functionality is not required, it can be switched off per PayGateID. By default, nobody is blind copied (Bcc) on payment confirmation emails, but if required a merchant may provide an email address which will be Bcc'd on each payment confirmation email sent.

PayVault: Default is Not Activated

Applies to: Card processing

PayVault is PayGate's credit card tokenisation service. When tokenisation of a credit card is requested the card's PAN and Expiry Date is stored in PayGate's PCI-compliant database and a PayVault 'token' (a GUID) is issued for the card. This token can then be re-used in place of the card number to process payments on that card via the PayGate system. By default, only credit cards for which the initial payment is approved will be added to the PayVault database and a token for the card issued. On request a terminal parameter can be set to allow for all cards to be added to the PayVault database and tokens issued, whether the initial payment is accepted or declined/failed.

Setup options when more than one payment method is activated

PayGate allows merchants to have multiple PayGateID's.

Each PayGate ID has access to the PayGate Back Office and all transactions processed by PayGate using a PayGate ID are visible in the back office. Reports can be viewed in the back office or downloaded into MS Excel (or similar) applications for offline reporting.

A merchant with multiple payment methods can choose to either:

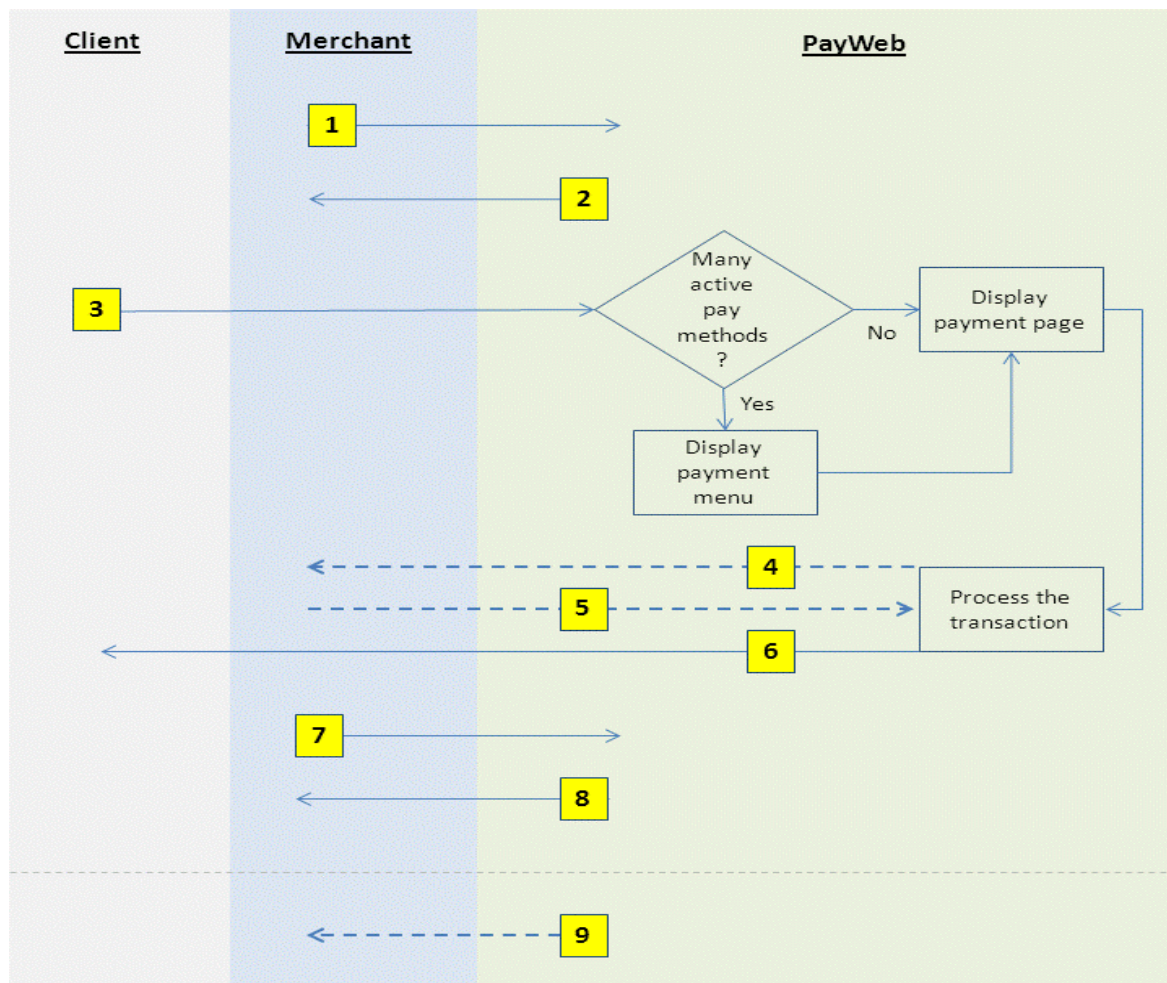
- a) have multiple payment methods all activated on a single PayGate ID or
- b) to have multiple PayGate ID's with a single payment method active per PayGate ID or
- c) to have multiple payment methods activated on a single PayGate ID and specify for each transaction which payment methods should be visible to the client (using the PAY_METHOD and PAY_METHOD_DETAIL fields to control this).

If option a) is chosen, then PayWeb will display a menu of payment options to the client. The client will choose how (s)he wants to pay and select the relevant menu option.

If option b) is chosen, then the client will be taken directly to the relevant payment page.

If option c) is chosen, then a menu of payment options will be shown only if more than one payment method meets the criteria specified in the PAY_METHOD and PAY_METHOD_DETAIL fields for the transaction.

Process flow diagram



Process flow description

1. The merchant begins the process by posting a detailed 'Request' to PayWeb.
2. PayWeb responds immediately with a unique PAY_REQUEST_ID field.
3. The merchant re-directs the client to PayWeb and passes limited information including the PAY_REQUEST_ID field returned by PayWeb in step 1.
PayWeb displays a menu of active payment methods to the client (if appropriate) and processes the transaction to the relevant financial service provider.
4. If the merchant places a value in the 'NOTIFY_URL' field in step 1 then PayWeb will post the 'Response' data back to the 'NOTIFY_URL' provided. This is done immediately, and before redirecting the client back to the 'RETURN_URL' in step 6.
5. If the NOTIFY_URL is specified by the merchant in step 1, then when PayWeb posts the 'Response' data to the NOTIFY_URL in step 4, the merchant must respond with 'OK' when the post is received.
6. PayWeb redirects the client back to the 'RETURN_URL' provided by the merchant in step 1.
7. The merchant can 'Query' PayWeb and post the PAY_REQUEST_ID field to PayWeb.
8. PayWeb replies immediately to step 6 by posting back detailed 'Response' data.

9. In some cases, the payment method chosen will be more suited to an 'asynchronous' process. For instance, when the client is given payment instructions by PayWeb and these instructions will take some time (possibly days) to complete. If/when PayWeb receives a response from the financial service provider stating that the transaction has been completed, then PayWeb will post the 'Response' data back to the 'NOTIFY_URL' provided by the merchant in step 1.

The Landing Pages

Payment menu page

The menu page is only displayed to the client if more than one payment method is activated on the PayGateID. Only active payment methods are displayed. This page is customisable and an iFrame can be used to maintain the look and feel of the merchant system as far as possible.



Example of a (non customised) payment menu page

Merchant	PAYGATE - TEST SYSTEM
Reference	Paygate Test
Transaction Date	Thu, 10 Mar 2016 10:49:16 +0200
Amount	R 5.00 (ZAR)

Please select a payment type

Credit Card

SiD Instant EFT

 Powered by  PayGate

Back

Payment page

The payment page will vary depending on the payment method.






This page is customisable and an iFrame can be used to maintain the look and feel of the merchant system as far as possible.

Example of a (non customised) credit card payment page

Merchant Reference	PAYGATE - TEST SYSTEM	
Transaction Date	Paygate Test	
Amount	Thu, 10 Mar 2016 10:49:16 +0200	
	R 5.00 (ZAR)	

Change Payment Type

Cards Accepted



Card Holder


Card Number

Expiry Date



▼

▼

CVV



☐ I've read and accept the Terms & Conditions

 Powered by  PayGate

Back

Next

Request and Response data

This section describes in detail the fields sent to PayWeb and the fields returned by PayWeb.

Please refer to the [process flow diagram](#) on page 7.

Step 1 – transaction request posted to PayWeb (the Request)

The merchant begins the process by posting a detailed 'Request' to PayWeb.

This post is generally done using cURL in PHP or HttpClient in .NET.

The URL to post this request to is: <https://secure.paygate.co.za/payweb3/initiate.trans>

Field	Type	Required
PAYGATE_ID Your PayGateID – assigned by PayGate	varchar(20)	Yes
REFERENCE This is your reference number for use by your internal systems	varchar(110)	Yes
AMOUNT Transaction amount in cents. e.g. 32.99 is specified as 3299	varchar(20)	Yes
CURRENCY Currency code of the currency the customer is paying in. Refer to Appendix A for valid currency codes	varchar(5)	Yes
RETURN_URL Once the transaction is completed, PayWeb will return the customer to a page on your web site. The page the customer must see is specified in this field	varchar(255)	Yes
TRANSACTION_DATE This is the date that the transaction was initiated on your website or system. The transaction date must be specified in 'Coordinated Universal Time' (UTC) e.g. 2016-05-30 09:30:10	dateTime	Yes
LOCALE The locale code identifies to PayGate the customer's language, country and any special variant preferences (such as Date/Time format) which may be applied to the user interface. Not all the locales in the locale table are supported by PayGate. Please confirm with support@paygate.co.za if the locale(s) you are using is supported. If the locale passed is not supported, then PayGate will default to the "en" locale	varchar(5)	Yes

COUNTRY Country code of the country the customer is paying from. Refer to Appendix A for valid country codes	varchar(5)	Yes
EMAIL If the transaction is approved, PayWeb will email a payment confirmation to this email address – unless this is overridden at a gateway level by using the Payment Confirmation setting. This field remains compulsory but the sending of the confirmation email can be blocked	varchar(255)	Yes
PAY_METHOD The payment method(s) to show to the client. <ul style="list-style-type: none"> If this field is not populated, then all payment methods activated will be shown on the menu page. If the merchant has more than one wallet method (EW) activated, and this field is populated with 'EW', then PayWeb will present the client with a menu of all the active wallet payment methods to choose from. If both the PAY_METHOD and PAY_METHOD_DETAIL fields are populated, then PayWeb will display the secure payment page for that specific payment method only. Refer to the Payment Method Codes table for a complete list	varchar(5)	No
PAY_METHOD_DETAIL The PAY_METHOD_DETAIL field should be left blank unless the merchant has more than one active payment method and wants to make sure that the client is presented with a specific payment method. Refer to the PAY_METHOD field above for more information	varchar(45)	No
NOTIFY_URL If the notify URL field is populated, then PayWeb will post the fields as specified in the Response table to the notify URL immediately when the transaction is completed. PayWeb will expect a response of 'OK'. If for any reason PayWeb cannot post to the notify URL successfully or if PayWeb doesn't receive the expected response of 'OK', then it will retry 3 times at 30 minute intervals before giving up. We only allow PORT 443 (HTTPS) secure and PORT 80 (HTTP) non-secure.	varchar(255)	No
USER1 This field is optional and has been included as a placeholder for merchant specific requirements. If this field is populated, then it must be included in the CHECKSUM calculation described below	varchar(255)	No
USER2 This field is optional and has been included as a placeholder for merchant specific requirements. If this field is populated, then it must be included in the CHECKSUM calculation described below	varchar(255)	No
USER3	varchar(255)	No

This field is optional and has been included as a placeholder for merchant specific requirements. If this field is populated, then it must be included in the CHECKSUM calculation described below		
VAULT This field is optional but should only be included if PayVault credit card tokenisation is enabled on the merchant profile. This field is used to indicate whether a PayVault token should be issued for the credit card used to make the payment. If True (1) the credit card number will be added to PayVault and the associated Token will be returned in the response to the merchant. 0 = false 1 = true	tinyint(3)	No
VAULT_ID This field is optional and should only be included if PayVault credit card tokenisation is enabled. If a PayVault token GUID is sent the credit card transaction will be processed using the credit card associated with the token. The cardholder will be shown the last 4 digits and expiry date of the credit card on the PayWeb page and will need to enter Cardholder Name and Credit Card CVV, as well as 3D Secure OTP if needed	varchar(40)	No
CHECKSUM This field contains a calculated MD5 hash based on the values of ALL the above-mentioned fields and a key . Refer to the section on Security below for more detail regarding the MD5 hash.	varchar(32)	Yes

Step 2 – Response to step 1 with PAY_REQUEST_ID

PayWeb responds immediately with a string, containing key/value pairs of data. This data will be used by the merchant's system in the following steps to collect response data securely from PayWeb.

Field	Type	Required
PAYGATE_ID This should be the same PayGate ID that was passed in the request; if it is not, then the data has been altered	varchar(20)	Yes
PAY_REQUEST_ID The PAY_REQUEST_ID is a GUID allocated by PayWeb to the transaction request received in step 1	varchar(32)	Yes
REFERENCE The reference that was passed in the step 1 request is returned unaltered	varchar(110)	Yes

CHECKSUM This field contains a calculated MD5 hash based on the values of ALL the above-mentioned fields and a key . Refer to the section on Security below for more detail regarding the MD5 hash	varchar(32)	Yes
--	-------------	-----

Step 3 – Re-direct client to PayWeb with PAY_REQUEST_ID

The merchant re-directs the client to the secure PayWeb payment page and passes only the PAY_REQUEST_ID and CHECKSUM fields.

All information posted to PayWeb should be placed in hidden form fields.

The HTML form element should resemble:

```
<form action="https://secure.paygate.co.za/payweb3/process.trans" method="POST" >
```

The URL for this re-direct is: <https://secure.paygate.co.za/payweb3/process.trans>

Field	Type	Required
PAY_REQUEST_ID The PAY_REQUEST_ID returned by PayWeb in step 2. e.g. <code><input type="hidden" name="PAY_REQUEST_ID" value="7B44FC55-CA90-1922-B32D-00DD010772DB"></code>	char(32)	Yes
CHECKSUM This field contains a calculated MD5 hash based on the values of the PAYGATE_ID, PAY_REQUEST_ID, REFERENCE fields and a key . Refer to the section on Security below for more detail regarding the MD5 hash. e.g. refer to the ' CHECKSUM example' for step 2 below	varchar(32)	Yes

Step 4 – Post ‘Response’ data back to NOTIFY_URL (optional)

If the merchant places a value in the ‘NOTIFY_URL’ field in step 1 then PayWeb will post the ‘Response’ data back to the ‘NOTIFY_URL’ provided. This is done immediately, and before re-directing the client back to the ‘RETURN_URL’ in step 5. (Please note, *we only allow PORT 443 (HTTPS) secure and PORT 80 (HTTP) non-secure*).

Field	Type	Required
PAYGATE_ID This should be the same PayGate ID that was passed in the request; if it is not, then the data has been altered.	varchar(20)	Yes
PAY_REQUEST_ID The PAY_REQUEST_ID returned by PayWeb in step 2.	char(32)	Yes
REFERENCE This should be the same reference that was passed in the request; if it is not, then the data has been altered.	varchar(110)	Yes
TRANSACTION_STATUS The final status of the transaction. Refer to the Transaction Status table for a list of possible values.	tinyint(3)	Yes
RESULT_CODE This field contains a code indicating the result of the transaction. Refer to the Result Code table for a complete list. The description corresponding to this code is in the RESULT_DESC field.	int(11)	Yes
AUTH_CODE If the bank or financial institution approves the transaction, then the authorisation code will be placed in this field. For non-card payment methods, this field is populated with “999999”.	varchar(10)	Yes
CURRENCY Currency code of the currency the customer is paying in. Refer to Appendix A for valid currency codes.	varchar(5)	Yes
AMOUNT This should be the same amount that was passed in the request. If it is not, then the data has been altered.	Number(11)	Yes
RESULT_DESC This field contains a description for the result of the transaction. Refer to the Result Code table for a complete list. The numeric code corresponding to this description is in the RESULT_CODE field.	int(11)	Yes

TRANSACTION_ID This field contains the PayGate unique reference number for the transaction.	int(11)	Yes
RISK_INDICATOR This is a 2-character field containing a risk indicator for this transaction. The first character describes the Verified-by-Visa / MasterCard SecureCode authentication; refer to the Authentication Indicator table for the possible values. The second character is for future use and will be set to 'X'. Please refer to the MasterCard SecureCode & Verified by Visa section for more info.	varchar(10)	Yes
PAY_METHOD This field contains a code describing/confirming the payment method used to process the transaction. It is especially useful where the merchant has more than one payment method activated. Refer to the Payment Method Codes table for a complete list.	varchar(5)	Yes
PAY_METHOD_DETAIL This field may contain a description of the PAY_METHOD code. For instance, if the PAY_METHOD is 'CC' to indicate credit card, then the PAY_METHOD_DETAIL will contain the type of credit card used 'MasterCard', 'Visa' etc. If the PAY_METHOD is something generic such as 'EW' = eWallet, then the PAY_METHOD_DETAIL field will contain the name of the eWallet. If populated, then this field is included in the CHECKSUM calculation described below.	varchar(45)	No
USER1 This field is optional and has been included as a placeholder for merchant specific requirements. If this field was populated in 'the Request' then the response will either contain the exact data sent in the Request or specific data as agreed with the merchant. If populated, then this field is included in the CHECKSUM calculation described below.	varchar(255)	No
USER2 This field is optional and has been included as a placeholder for merchant specific requirements. If this field was populated in 'the Request' then the response will either contain the exact data sent in the Request or specific data as agreed with the merchant. If populated, then this field is included in the CHECKSUM calculation described below.	varchar(255)	No
USER3 This field is optional and has been included as a placeholder for merchant specific requirements. If this field was populated in 'the Request' then the response will either contain the exact data sent in the Request or specific data as agreed with the merchant. If populated, then this field is included in the CHECKSUM calculation described below.	varchar(255)	No

VAULT_ID This is the PayVault token associated to the card used to make the payment. This Vault ID can be re-used to process payments on the card either via PayWeb or PayBatch. Only the PAN and Expiry Date are linked to this token. This is an optional field and is only returned if PayVault tokenisation is requested.	varchar (40)	No
PAYVAULT_DATA_1 This field contains information on the credit card or e-wallet account linked to the PayVault token for managing the use of the token. This is an optional field and is only returned if PayVault tokenisation is requested.	varchar(50)	No
PAYVAULT_DATA_2 This field contains information on the credit card or e-wallet account linked to the PayVault token for managing the use of the token. This is an optional field and is only returned if PayVault tokenisation is requested.	varchar(50)	No
CHECKSUM This field contains a calculated MD5 hash based on the values of ALL the above-mentioned fields and a key . Refer to the section on Security below for more detail regarding the MD5 hash. Refer to the EXAMPLES for an example of this calculation.	varchar(32)	Yes

Step 5 – Merchant responds with ‘OK’

If the NOTIFY_URL is specified by the merchant in step 1, then when PayWeb posts the ‘Response’ data to the NOTIFY_URL in step 4, the merchant must respond with ‘OK’ when the post is received. (Please note, we only allow *PORT 443 (HTTPS) secure and PORT 80 (HTTP) non-secure*).

There is no CHECKSUM and no other data fields are sent.

Step 6 – Re-direct the client back to the merchant’s web site

PayWeb redirects the client back to the ‘RETURN_URL’ provided by the merchant in step 1.

Field	Type	Required
PAY_REQUEST_ID The PAY_REQUEST_ID returned by PayWeb in step 2	varchar(32)	Yes
TRANSACTION_STATUS The final status of the transaction. Refer to the Transaction Status table for a list of possible values.	tinyint(3)	Yes

CHECKSUM This field contains a calculated MD5 hash based on the values of the PAYGATE_ID, PAY_REQUEST_ID, TRANSACTION_STATUS, REFERENCE fields and a key . Refer to the section on Security below for more detail regarding the MD5 hash. Refer to the EXAMPLES for an example of this calculation.	varchar(32)	Yes
---	-------------	-----

Step 7 – Merchant posts PAY_REQUEST_ID to PayWeb

The merchant posts the PAY_REQUEST_ID field to PayWeb to retrieve the detailed response data.

This step is optional in that it does not affect the result of the transaction in any way, but it provides the merchant with a mechanism to retrieve detailed response data. This step can be done many times (if necessary) and can be used by the merchant to query transaction response data at any time.

PayGate will make this data available for at least 6 months from the date of the transaction.

The request is done the same way that the Initiate step is done.

The URL to post this query to is: <https://secure.paygate.co.za/payweb3/query.trans>

Field	Type	Required
PAYGATE_ID Your PayGateID – assigned by PayGate	varchar(20)	Yes
PAY_REQUEST_ID The PAY_REQUEST_ID returned by PayWeb in step 2	varchar(32)	Yes
REFERENCE The REFERENCE field passed to PayWeb in step1	varchar(110)	Yes
CHECKSUM This field contains a calculated MD5 hash based on the values of the PAYGATE_ID, PAY_REQUEST_ID, REFERENCE fields and a key . Refer to the section on Security below for more detail regarding the MD5 hash. Refer to the EXAMPLES for an example of this calculation.	varchar(32)	Yes

Step 8 – PayWeb Posts ‘Response’ data back to the request in step 7

When PayWeb receives a post containing a valid PAY_REQUEST_ID then the relevant transaction ‘Response’ data is posted back immediately.

Field	Type	Required
The field data returned by step 8 is identical to that returned by step 4.		

Step 9 – Final transaction notification posted to NOTIFY_URL

In some cases, the payment method chosen will be more suited to an ‘asynchronous’ process. For instance, when the client is given payment instructions by PayWeb and these instructions will take some time (possibly days) to complete. If/when PayWeb receives a response from the financial service provider stating that the transaction has been completed (or that the transaction status has changed), then PayWeb will post the ‘Response’ data back to the ‘NOTIFY_URL’ provided by the merchant in step 1.

The NOTIFY_URL must return the value ‘OK’ to indicate that the post was received. If PayWeb cannot contact the merchant’s NOTIFY_URL and/or if an ‘OK’ reply is not received, then PayWeb will try twice more at 30 minute intervals before giving up. (Please note, *we only allow PORT 443 (HTTPS) secure and PORT 80 (HTTP) non-secure*).

Field	Type	Required
The field data returned by step 9 is identical to that returned by step 4.		

Miscellaneous Information

Security

Security is enhanced by making use of an MD5 checksum value that is passed in both the request to PayWeb and the response from PayWeb.

The checksum in all cases is calculated by concatenating all the fields in the relevant ‘step’ even if the field is optional.

An Encryption Key is appended and the resulting string is passed through an MD5 hash algorithm to produce the checksum. When PayGate receives the PayWeb request, the same checksum calculation is performed. If the PayGate checksum does not match the checksum specified in the request, the transaction is rejected.

The merchant must do the checksum calculation when a response is received from PayWeb. If the calculated checksum does not match the PayGate checksum in the response, the results should be rejected.

MD5 is a one-way hashing algorithm. Simply stated, input of any length supplied to the function produces a fixed length (in this case 32 characters) output so that the original input is not recognizable. It is impossible to reverse; i.e. giving the function the result will not give you the original source. Most programming languages support the MD5 function; if not native support then by a module or extension. You can find more information on MD5 implementation in various programming languages at the website:
<http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html>.

The Encryption Key used in the checksum calculation should only be known by the merchants' website and PayGate. It should not be displayed on any web page i.e. a customer should never be able to see it. PayGate allows for the Encryption Key to be a maximum of 32 alphanumeric characters. The longer and more complex the key is, the harder it is for a malicious user to guess it.

CHECKSUM examples

The **key** is only known by the merchant and PayGate (via the PayGate BackOffice) and should not be displayed on the merchant's web site. PayWeb does the same calculation when the request is received to ensure that the data has not been tampered with.

Step 1 - Checksum Examples

All fields including the optional fields are concatenated (there is no separator character) to form the source of the MD5 hash:

```
PAYGATE_ID+REFERENCE+AMOUNT+CURRENCY+RETURN_URL+TRANSACTION_DATE+LOCALE+COUNTRY+EMAIL  
+PAY_METHOD+PAY_METHOD_DETAIL+NOTIFY_URL+USER1+USER2+USER3+VAULT+VAULT_ID+KEY
```

Assuming the **KEY** is 'secret', the following scenarios are possible:

1. If all the optional fields are empty, the checksum source would translate to:

```
10011072130PayGate Test3299ZARhttps://www.paygate.co.za/thankyou2016-03-10 10:49:16enZAFcustomer@paygate.co.zasecret  
The MD5 hash value for this transaction would be: 0bcaea6fa6bc0337e066db9826088557  
<input type="hidden" name="CHECKSUM" value=" 0bcaea6fa6bc0337e066db9826088557">
```

2. With the **NOTIFY_URL** and **USER1** fields populated, the checksum source would translate to:

```
10011072130PayGate Test3299ZARhttps://www.paygate.co.za/thankyou2016-03-10  
10:49:16enZAFcustomer@paygate.co.zahttps://www.paygate.co.za/notifySpecialKeysecret  
The MD5 hash value for this transaction would be: df991fae434fdb29c9135ef0baac1194
```

Step 2 - Checksum Example

All fields are concatenated (there is no separator character) to form the source of the MD5 hash:

```
PAYGATE_ID+PAY_REQUEST_ID+REFERENCE+KEY
```

Assuming the **KEY** is 'secret', the following scenario is possible:

```
1001107213026F1EE9D-FB68-D6C2-5D36-ADA8C5F88BC9PayGate Testsecret  
The MD5 hash value for this transaction would be: f5563213b72cb405167ba53e8c3ee466
```

Step 3 - Checksum Example

Concatenate the PAYGATE_ID, PAY_REQUEST_ID, REFERENCE AND KEY (no separator characters) to form the source of the MD5 hash:

PAYGATE_ID+PAY_REQUEST_ID+REFERENCE+KEY

Assuming the **KEY** is 'secret', the following scenario is possible:

1001107213026F1EE9D-FB68-D6C2-5D36-ADA8C5F88BC9PayGate Testsecret

The MD5 hash value for this transaction would be: f5563213b72cb405167ba53e8c3ee466

<input type="hidden" name="CHECKSUM" value=" f5563213b72cb405167ba53e8c3ee466 ">

Step 4 - Checksum Examples

All fields including the optional fields are concatenated (there is no separator character) to form the source of the MD5 hash:

PAYGATE_ID+PAY_REQUEST_ID+REFERENCE+TRANSACTION_STATUS+RESULT_CODE+AUTH_CODE+CURRENCY
+AMOUNT+RESULT_DESC+TRANSACTION_ID+RISK_INDICATOR+PAY_METHOD+PAY_METHOD_DETAIL+USER1
+USER2+USER3+VAULT_ID+CARD_USED+EXPIRY_DATE+KEY

Assuming the KEY is 'secret' and the USER fields are not populated, the checksum source would translate to:

1001107213026F1EE9D-FB68-D6C2-5D36-ADA8C5F88BC9PayGate Test1990017P3TPSQZAR3299Auth Done36645089AXCCVisasecret

The MD5 hash value for this transaction would be: 53e1561ed2b98db6221b3f0c387a0770

Assuming the KEY is 'secret' and the USER1 field is populated, the checksum source would translate to:

1001107213026F1EE9D-FB68-D6C2-5D36-ADA8C5F88BC9PayGate Test1990017P3TPSQZAR3299Auth
Done36645089AXCCVisaSpecialKeysecret

The MD5 hash value for this transaction would be: 56310d6fdab5561cce43620842c63dd6

Step 5 – No Checksum

No checksum is required for step 5.

Step 6 - Checksum Example

Concatenate the PAYGATE_ID, PAY_REQUEST_ID, TRANSACTION_STATUS, REFERENCE AND KEY (no separator characters) to form the source of the MD5 hash:

PAYGATE_ID+PAY_REQUEST_ID+TRANSACTION_STATUS+REFERENCE+KEY

Assuming the **KEY** is 'secret', the following scenario is possible:

1001107213026F1EE9D-FB68-D6C2-5D36-ADA8C5F88BC91PayGate Testsecret

The MD5 hash value for this transaction would be: 2fae4c5cde9ac8ed70f769c3ff843d72

Step 7 - Checksum Example

Concatenate the PAYGATE_ID, PAY_REQUEST_ID, REFERENCE AND KEY (no separator characters) to form the source of the MD5 hash:

PAYGATE_ID+PAY_REQUEST_ID+REFERENCE+KEY

Assuming the **KEY** is 'secret', the following scenario is possible:

1001107213026F1EE9D-FB68-D6C2-5D36-ADA8C5F88BC9PayGate Testsecret

The MD5 hash value for this transaction would be: f5563213b72cb405167ba53e8c3ee466

Request and Response examples

Step 1 - Request Examples

These examples assume the Encryption Key '**secret**' was used as part of the **CHECKSUM** calculation.

A PHP Example Without any of the optional fields:

```
//set the data
$data = array (
    'PAYGATE_ID'      => '10011072130',
    'REFERENCE'       => 'PayGate Test',
    'AMOUNT'          => '3299',
    'CURRENCY'        => 'ZAR',
    'RETURN_URL'      => 'https://www.paygate.co.za/thankyou',
    'TRANSACTION_DATE' => '2016-03-24 14:53:15',
    'LOCALE'          => 'en',
    'COUNTRY'         => 'ZAF',
    'EMAIL'           => 'customer@paygate.co.za',
    'CHECKSUM'        => 'd542d24ad60c422274002457fdb7397e'
);
$postData = http_build_query($data);
//open connection
$ch = curl_init();
//set the url, number of POST vars, POST data
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt($ch, CURLOPT_URL, 'https://secure.paygate.co.za/payweb3/initiate.trans');
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_NOBODY, false);
curl_setopt($ch, CURLOPT_REFERER, $_SERVER['HTTP_HOST']);
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $postData);

//execute post
$result = curl_exec($ch);
//close connection
curl_close($ch);
```

A PHP Example containing the NOTIFY_URL and USER1 fields:

```
//set the data
$data = array (
    'PAYGATE_ID'      => '10011072130',
    'REFERENCE'       => 'PayGate Test',
    'AMOUNT'          => '3299',
    'CURRENCY'        => 'ZAR',
    'RETURN_URL'      => 'https://www.paygate.co.za/thankyou',
    'TRANSACTION_DATE' => '2016-03-24 14:53:15',
    'LOCALE'          => 'en',
    'COUNTRY'         => 'ZAF',
    'EMAIL'           => 'customer@paygate.co.za',
    'NOTIFY_URL'      => 'https://www.paygate.co.za/notify',
    'USER1'           => 'SpecialKey',
    'CHECKSUM'        => 'd542d24ad60c422274002457fdb7397e'
);

$postData = http_build_query($data);

//open connection
$ch = curl_init();
//set the url, number of POST vars, POST data
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
```



```
curl_setopt($ch, CURLOPT_URL, 'https://secure.paygate.co.za/payweb3/initiate.trans');
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_NOBODY, false);
curl_setopt($ch, CURLOPT_REFERER, $_SERVER['HTTP_HOST']);
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $postData);
//execute post
$result = curl_exec($ch);
//close connection
curl_close($ch);
```

Step 4 - Response Example

The following examples assumes the Encryption Key '**secret**' was used as part of the **CHECKSUM** calculation.

A PHP Example of an Initiate Response:

The `$result = curl_exec($ch);` will contain the following data:

```
array [
    "PAYGATE_ID"      => "10011072130",
    "PAY_REQUEST_ID"  => "255A1560-5B66- 1BB5-CD64- B5DC96F381FD",
    "REFERENCE"       => "PayGate Test",
    "CHECKSUM"        => "514a4028b8dca379dfc979132e0a0cc6"
]
```

Redirect Back to Merchant:

The `$result = curl_exec($ch);` will contain the following data:

```
array [
    "PAY_REQUEST_ID"      => "255A1560-5B66- 1BB5-CD64- B5DC96F381FD",
    "TRANSACTION_STATUS" => "1",
    "CHECKSUM"           => "514a4028b8dca379dfc979132e0a0cc6"
]
```

Notify Response:

The `$result = curl_exec($ch);` will contain the following data:

```
array [
    "PAYGATE_ID"      => "10011072130",
    "PAY_REQUEST_ID"  => "255A1560-5B66-1BB5-CD64-B5DC96F381FD",
    "REFERENCE"       => "PayGate Test",
    "TRANSACTION_STATUS" => "1",
    "RESULT_CODE"     => "990017",
    "AUTH_CODE"       => "BUH8UE",
    "CURRENCY"        => "ZAR",
    "AMOUNT"          => "3299",
    "RESULT_DESC"     => "Auth Done",
    "TRANSACTION_ID"  => "36969389",
    "RISK_INDICATOR"  => "AX",
    "PAY_METHOD"      => "CC",
    "PAY_METHOD_DETAIL" => "Visa",
    "CHECKSUM"        => "ef23cd49953b459e603f34f568a45892"
]
```

This PHP example assumes that the USER1 field was populated with 'SpecialKey' in the Request.

The `$result = curl_exec($ch);` will contain the following data:

```
array [  
    "PAYGATE_ID"          => "10011072130",  
    "PAY_REQUEST_ID"     => "255A1560-5B66-1BB5-CD64-B5DC96F381FD",  
    "REFERENCE"          => "PayGate Test",  
    "TRANSACTION_STATUS" => "1",  
    "RESULT_CODE"        => "990017",  
    "AUTH_CODE"          => "BUH8UE",  
    "CURRENCY"           => "ZAR",  
    "AMOUNT"             => "3299",  
    "RESULT_DESC"        => "Auth Done",  
    "TRANSACTION_ID"     => "36969389",  
    "RISK_INDICATOR"     => "AX",  
    "PAY_METHOD"         => "CC",  
    "PAY_METHOD_DETAIL"  => "Visa",  
    "USER1"              => "SpecialKey",  
    "CHECKSUM"           => "b5fd70d991acb5d057bb1d9b17992010"  
]
```

Testing

For testing please use the following credentials:

PayGate ID: **10011072130**

Encryption Key: **secret**

Testing Currencies: **ZAR, EUR, USD**

All requests using this PayGate ID are processed to a transaction simulator on our production system.

Please refer to the table below when testing to simulate predictable results:

Card Brand	Card Number	Risk Indicator
Approved Transactions. RESULT_CODE = 990017; TRANSACTION_STATUS = 1.		
Visa	4000000000000002	Authenticated (AX) *
MasterCard	5200000000000015	Authenticated (AX)
American Express	378282246310005	Not Authenticated (NX)
M-Pesa	N/A; enter MR PASS in First & Last Name field.	Authenticated (AX)
Insufficient Funds Transactions. RESULT_CODE = 900003; TRANSACTION_STATUS = 2.		
MasterCard	5200000000000023	Not Authenticated (NX) *
Visa	4000000000000028	Not Authenticated (NX)
American Express	371449635398431	Not Authenticated (NX)
Declined Transactions. RESULT_CODE = 900007; TRANSACTION_STATUS = 2.		
Visa	4000000000000036	Authenticated (AX) *
MasterCard	5200000000000049	Authenticated (AX) *
Diners Club	30569309025904	Not Applicable (XX)
M-Pesa	N/A; enter MR FAIL in First & Last Name field	Not Applicable (XX)
Invalid Card Number. RESULT_CODE = 900004; TRANSACTION_STATUS = 2.		
For credit card payment method - all other card numbers		Not Applicable (XX)
Unprocessed Transactions. RESULT_CODE = 990022; TRANSACTION_STATUS = 0.		
MasterCard	5200000000000064	Not Applicable (XX)
<i>Expiry Date must be in the future; Card Holder & CVV can be made up.</i>		

* Using these card numbers will allow you to test the MasterCard SecureCode / Verified-by-Visa authentication process.

MasterCard SecureCode & Verified by Visa

What is Secure Code and Verified by Visa?

3D Secure is a MasterCard and Visa initiative to reduce online credit card transaction fraud and applies to Visa and MasterCard transactions only.

The Visa implementation is referred to as Verified by Visa or V-by-V.

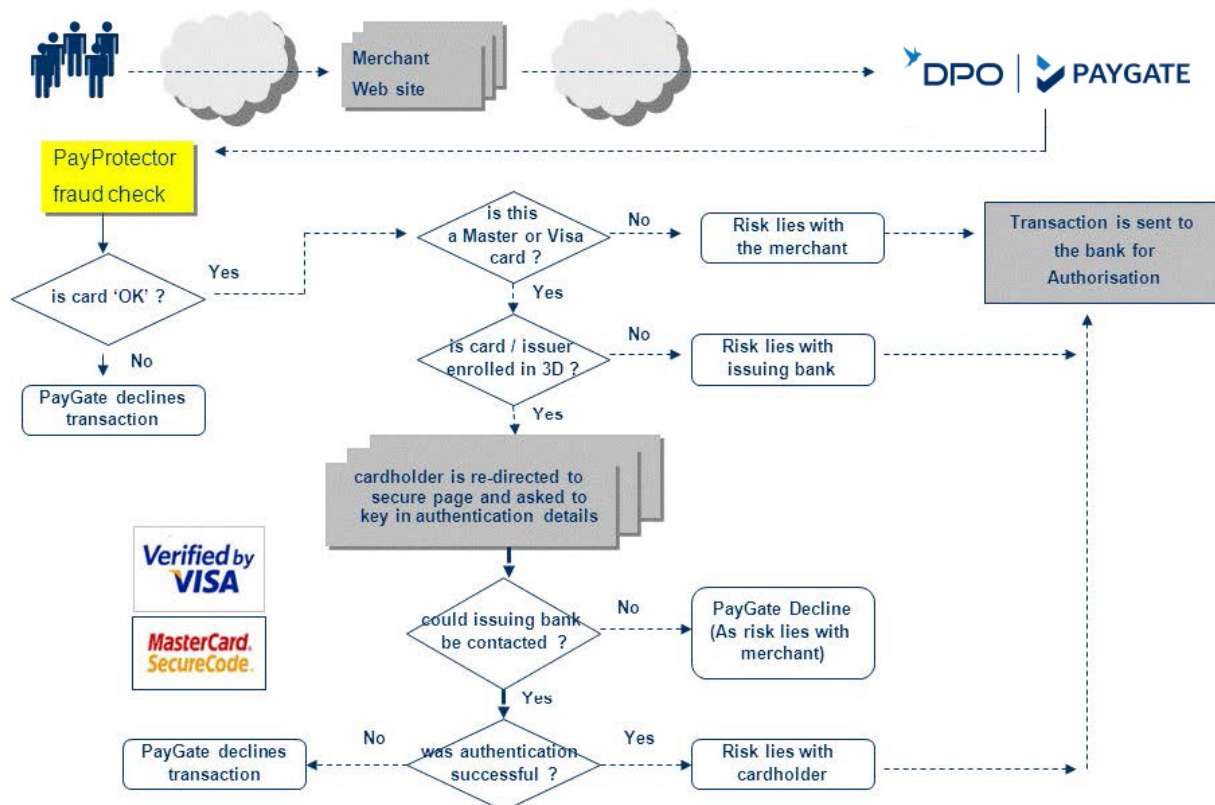
The MasterCard implementation is referred to as MasterCard Secure Code.

How does Secure Code and Verified by Visa benefit the merchant?

It significantly reduces the risk of fraudulent transactions, and moves the risk of certain charge backs from the merchant to the card holder or the Issuing Bank.

(Note – there are instances where the charge back risk remains with the merchant – this is detailed in the flowchart below).

A typical credit card authorisation flow including PayProtector and 3D



How Does Secure Code and Verified by Visa work?

When a purchase is made online, the cardholder will be redirected from the secure PayGate payment page, to the issuing bank's (cardholder's bank) SecureCode and Verified by Visa authentication page. Here the cardholder will be required to key in his/her authentication details (e.g. a one-time PIN sent in a message to their mobile phone).

The Issuing Bank validates this code and returns an 'OK' or 'not OK' response to PayGate. If PayGate receives an 'OK' response, then we pass the transaction on to the Acquiring Bank for Authorisation. If the response is 'not OK' then the transaction is 'Declined' up front by PayGate.

It should however be noted that not all Issuing Banks will force their cardholders to register for this service. Where this is the case, 3D Secure authentication will still be attempted but the card holder will not be required to enter a PIN or password by their bank. A message will be returned to PayGate to indicate that you as a merchant attempted to authenticate the transaction and that the issuing bank is not registered for the service. The transaction will be processed as a SecureCode and Verified by Visa transaction i.e. the risk will be passed to the issuing bank.

What about the other cards (AMEX, Diners etc)?

These cards are not authenticated via the Secure Code and Verified by Visa process. Now transaction risk for purchases made with cards other than Master and Visa, will remain with the merchant.

Frequently Asked Questions

How do I know the transaction is approved?

You can check up to 3 fields in the response depending on how thorough you want to be. At a minimum, you should check the TRANSACTION_STATUS field: it will contain the value "1". If you want to check further, the RESULT_CODE field should contain the value "990017" and the AUTH_CODE field should not be blank.

Can I do the authorisation and the settlement separately?

Yes, PayGate has an 'Auto-Settle' configuration setting that is enabled by default for all merchants. This means that PayGate automatically creates the settlement transaction when a PayWeb request is approved. If you would prefer to only authorise the transaction when the customer enters their card details (i.e. no funds are transferred), please send an email to support@paygate.co.za to request that the 'Auto-Settle' feature be disabled. With the 'Auto-Settle' feature disabled, the merchant must login to the PayGate BackOffice and effect the settlement manually.

What response is returned if the customer clicks the 'Cancel' button on the PayWeb payment page?

- The TRANSACTION_STATUS field will contain "3".
- The RESULT_CODE field will contain "990028".
- The TRANSACTION_ID field will be blank.

How will I know that the transaction was authenticated and I have charge back protection?

When your website receives the transaction results from PayGate, it should check the first character of the RISK_INDICATOR field. If the first character is 'A' then your customer has been authenticated and cannot initiate a charge back. If the first character is 'N' then the transaction has been declined or approved but not authenticated; you should take further steps to ensure that you are dealing with the legitimate card holder.

The transaction was authenticated and declined; how can this be?

PayGate attempts to authenticate the cardholder before sending the transaction to the bank for authorisation. Therefore, even if the cardholder is authenticated through MasterCard SecureCode or Verified-by-Visa, the bank could still decline the transaction due to insufficient funds etc.

Is it possible to not use Verified-by-Visa and MasterCard SecureCode?

3D Secure is mandatory for e-commerce transactions in many countries and acquiring banks may only issue acquiring accounts that have been 3D Secure registered. PayGate can de-activate 3D Secure on a merchant profile level only with express permission to do so from the merchant's bank.

PayVault Validation Information

If a VAULT_ID PayVault Token is sent in the PayWeb Request, then for a transaction to be processed:

- PayVault must be enabled on the PayGate account.
- The VAULT_ID must be valid for the PAY_METHOD passed in the request or, if no PAY_METHOD is passed, the VAULT_ID must be valid for one of the payment methods available on the PayGate account.

The PayVault Token passed as the VAULT_ID must be an active (not deleted) and cannot be expired for the relevant payment method (e.g. for an expired credit card). Validation will fail the request if an invalid or expired Token is sent.

If there are multiple payment methods active on a PayGate account and a VAULT_ID is sent in a request without a PAY_METHOD being specified then the payment method associated with the PayVault Token will be displayed as the default method, but the user will be given the option to change payment method on the PayWeb page.

PayVault tokenisation will only take place when:

- PayVault is enabled on the merchant's PayGate profile.
- A VAULT value of '1' is sent in the Request. If no VAULT value is sent it is assumed to be '0' and no tokenisation will take place.
- PayVault tokenisation is supported for the payment method selected by the user if multiple payment methods are available. Currently only credit card (PAY_METHOD of 'CC') is supported by PayVault for tokenisation.
- The payment is approved by the acquirer. This is set as default in the PayGate account configuration but can be set to tokenise all transactions whether approved or not.

Appendix A : Codes & Descriptions

Result Codes

Code	Description	Comment
Credit Card Errors – These RESULT_CODES are returned if the transaction cannot be authorised due to a problem with the card. The TRANSACTION_STATUS will be 2 .		
900001	Call for Approval	
900002	Card Expired	
900003	Insufficient Funds	
900004	Invalid Card Number	
900005	Bank Interface Timeout	Indicates a communications failure between the banks systems.
900006	Invalid Card	
900007	Declined	
900009	Lost Card	
900010	Invalid Card Length	
900011	Suspected Fraud	
900012	Card Reported as Stolen	
900013	Restricted Card	
900014	Excessive Card Usage	
900015	Card Blacklisted	
900207	Declined; authentication failed	Indicates the cardholder did not enter their MasterCard SecureCode / Verified by Visa password correctly.
990020	Auth Declined	
900210	3D Secure Lookup Timeout	
991001	Invalid expiry date	
991002	Invalid Amount	
Transaction Successful – Indicates the transaction was approved. TRANSACTION_STATUS will be 1 .		
990017	Auth Done	
Communication Errors – These RESULT_CODES are returned if the transaction cannot be completed due to an unexpected error. TRANSACTION_STATUS will be 0 .		
900205	Unexpected authentication result (phase 1)	
900206	Unexpected authentication result (phase 2)	
990001	Could not insert into Database	
990022	Bank not available	
990053	Error processing transaction	
Miscellaneous - Unless otherwise noted, the TRANSACTION_STATUS will be 0 .		
900209	Transaction verification failed (phase 2)	Indicates the verification data returned from MasterCard SecureCode / Verified-by-Visa has been altered.

900210	Authentication complete; transaction must be restarted	Indicates that the MasterCard SecureCode / Verified-by-Visa transaction has already been completed. Most likely caused by a customer clicking the refresh button.
900019	Invalid PayVault Scope.	
990024	Duplicate Transaction Detected. Please check before submitting	
990028	Transaction cancelled	Customer clicks the 'Cancel' button on the payment page.

Transaction Status

Transaction Code	Description
0	Not Done
1	Approved
2	Declined
3	Cancelled
4	User Cancelled

MasterCard SecureCode / Verified by Visa Authentication Indicator

Code	Description	Comment
N	Not Authenticated	Authentication was attempted but NOT successful. Merchant does NOT receive charge back protection for this transaction.
A	Authenticated	Authentication was attempted and was successful. Merchant does receive charge back protection for this transaction.
X	Not Applicable	Authentication processing NOT enabled on PayGate account or unexpected error in authentication process. Merchant does NOT receive charge back protection for this transaction.

Payment Method Codes

Pay Method	Description
CC	Credit Card
DC	Debit Card
EW	eWallet
BT	Bank Transfer
CV	Cash Voucher
PC	Pre-Paid Card

Locale Codes

Af	Afrikaans	Sq	Albanian
ar-sa	Arabic (Saudi Arabia)	ar-iq	Arabic (Iraq)
ar-eg	Arabic (Egypt)	ar-ly	Arabic (Libya)
ar-dz	Arabic (Algeria)	ar-ma	Arabic (Morocco)
ar-tn	Arabic (Tunisia)	ar-om	Arabic (Oman)
ar-ye	Arabic (Yemen)	ar-sy	Arabic (Syria)
ar-jo	Arabic (Jordan)	ar-lb	Arabic (Lebanon)
ar-kw	Arabic (Kuwait)	ar-ae	Arabic (U.A.E.)
ar-bh	Arabic (Bahrain)	ar-qa	Arabic (Qatar)
Eu	Basque	bg	Bulgarian

Be	Belarusian	ca	Catalan
zh-tw	Chinese (Taiwan)	zh-cn	Chinese (PRC)
zh-hk	Chinese (Hong Kong SAR)	zh-sg	Chinese (Singapore)
Hr	Croatian	cs	Czech
Da	Danish	nl	Dutch (Standard)
nl-be	Dutch (Belgium)	en	English
en-us	English (United States)	en-gb	English (United Kingdom)
en-au	English (Australia)	en-ca	English (Canada)
en-nz	English (New Zealand)	en-ie	English (Ireland)
en-za	English (South Africa)	en-jm	English (Jamaica)
En	English (Caribbean)	en-bz	English (Belize)
en-tt	English (Trinidad)	et	Estonian
fo	Faeroese	fa	Farsi
fi	Finnish	fr	French (Standard)
fr-be	French (Belgium)	fr-ca	French (Canada)
fr-ch	French (Switzerland)	fr-lu	French (Luxembourg)
gd	Gaelic (Scotland)	ga	Irish
de	German (Standard)	de-ch	German (Switzerland)
de-at	German (Austria)	de-lu	German (Luxembourg)
de-li	German (Liechtenstein)	el	Greek
he	Hebrew	hi	Hindi
hu	Hungarian	is	Icelandic
id	Indonesian	it	Italian (Standard)
it-ch	Italian (Switzerland)	ja	Japanese
ko	Korean	ko	Korean (Johab)
lv	Latvian	lt	Lithuanian
mk	Macedonian (FYROM)	ms	Malaysian
mt	Maltese	no	Norwegian (Bokmal)
no	Norwegian (Nynorsk)	pl	Polish
pt-br	Portuguese (Brazil)	pt	Portuguese (Portugal)
rm	Rhaeto-Romanic	ro	Romanian
ro-mo	Romanian (Republic of Moldova)	ru	Russian
ru-mo	Russian (Republic of Moldova)	sz	Sami (Lappish)

sr	Serbian (Cyrillic)	sr	Serbian (Latin)
sk	Slovak	sl	Slovenian
sb	Sorbian	es	Spanish (Spain)
es-mx	Spanish (Mexico)	es-gt	Spanish (Guatemala)
es-cr	Spanish (Costa Rica)	es-pa	Spanish (Panama)
es-do	Spanish (Dominican Republic)	es-ve	Spanish (Venezuela)
es-co	Spanish (Colombia)	es-pe	Spanish (Peru)
es-ar	Spanish (Argentina)	es-ec	Spanish (Ecuador)
es-cl	Spanish (Chile)	es-uy	Spanish (Uruguay)
es-py	Spanish (Paraguay)	es-bo	Spanish (Bolivia)
es-sv	Spanish (El Salvador)	es-hn	Spanish (Honduras)
es-ni	Spanish (Nicaragua)	es-pr	Spanish (Puerto Rico)
sx	Sutu	sv	Swedish
sv-fi	Swedish (Finland)	th	Thai
ts	Tsonga	tn	Tswana
tr	Turkish	uk	Ukrainian
ur	Urdu	ve	Venda
vi	Vietnamese	xh	Xhosa
ji	Yiddish	zu	Zulu

Country and Currency codes

Country	Country Code	Currency	Currency Code
Afghanistan	AFG	Afghani	AFA
Albania	ALB	Lek	ALL
Algeria	DZA	Algerian Dinar	DZD
American Samoa	ASM	U.S. Dollar	USD
Andorra	AND	Euro	EUR
Angola	AGO	Kwanza	AOA
Anguilla	AIA	E. Caribbean Dollar	XCD
Antarctica	ATA	Norwegian Krone	NOK
Antigua and Barbuda	ATG	E. Caribbean Dollar	XCD
Argentina	ARG	Argentine Peso	ARS
Armenia	ARM	Armenian Dram	AMD
Aruba	ABW	Aruban Guilder	AWG

Australia	AUS	Australian Dollar	AUD
Austria	AUT	Euro	EUR
Azerbaijan	AZE	Azerbaijan Manat	AZM
Bahamas	BHS	Bahamian Dollar	BSD
Bahrain	BHR	Bahraini Dinar	BHD
Bangladesh	BGD	Taka	BDT
Barbados	BRB	Barbados Dollar	BBD
Belarus	BLR	Belarussian Ruble	BYR
Belgium	BEL	Euro	EUR
Belize	BLZ	Belize Dollar	BZD
Benin	BEN	CFA Franc BCEAO	XOF
Bermuda	BMU	Bermudian Dollar	BMD
Bhutan	BTN	Indian Rupee	INR
Bolivia	BOL	Boliviano	BOB
Bosnia and Herzegovina	BIH	Bosnian Convertible Mark	BAM
Botswana	BWA	Pula	BWP
Bouvet Is.	BVT	Norwegian Krone	NOK
Brazil	BRA	Brazilian Real	BRL
British Indian Ocean Territory	IOT	U.S. Dollar	USD
British Virgin Is.	VGB	U.S. Dollar	USD
Brunei Darussalam	BRN	Brunei Dollar	BND
Bulgaria	BGR	Bulgarian Lev	BGN
Burkina Faso	BFA	CFA Franc BCEAO	XOF
Burundi	BDI	Burundi Franc	BIF
Cambodia	KHM	Riel	KHR
Cameroon United Republic of	CMR	CFA Franc BEAC	XAF
Canada	CAN	Canadian Dollar	CAD
Cape Verde Is.	CPV	Cape Verde Escudo	CVE
Cayman Is.	CYM	Cayman Is. Dollar	KYD
Central African Republic	CAF	CFA Franc BEAC	XAF
Chad	TCD	CFA Franc BEAC	XAF
Chile	CHL	Chilean Peso	CLP
China	CHN	Yuan Renminbi	CNY
Christmas Is.	CXR	Australian Dollar	AUD
Cocos (Keeling) Is.	CCK	Australian Dollar	AUD
Colombia	COL	Colombian Peso	COP
Comoros	COM	Comoro Franc	KMF
Congo	COG	CFA Franc BEAC	XAF

Cook Is.	COK	New Zealand Dollar	NZD
Costa Rica	CRI	Costa Rican Colon	CRC
Côte d'Ivoire (Ivory Coast)	CIV	CFA Franc BCEAO	XOF
Croatia	HRV	Croatian Kuna	HRK
Cuba	CUB	Cuban Peso	CUP
Cyprus	CYP	Cyprus Pound	CYP
Czech Republic	CZE	Czech Koruna	CZK
Democratic Republic of the Congo (formerly Zaire)	COD	Franc Congolais (formerly New Zaire)	CDF
Denmark	DNK	Danish Krone	DKK
Djibouti	DJI	Djibouti Franc	DJF
Dominica	DMA	E. Caribbean Dollar	XCD
Dominican Rep.	DOM	Dominican Peso	DOP
East Timor	TMP	Timor Escudo	TPE
Ecuador	ECU	Sucre	ECS
Egypt	EGY	Egyptian Pound	EGP
El Salvador	SLV	U.S. Dollar	USD
Equatorial Guinea	GNQ	CFA Franc BEAC	XAF
Eritrea	ERI	Eritrean Nakfa	ERN
Estonia	EST	Kroon	EEK
Ethiopia	ETH	Ethiopian Birr	ETB
European Monetary Cooperation Fund	--	European Currency Unit	XEU
European Union	--	Euro	EUR
Faeroe Is.	FRO	Danish Krone	DKK
Falkland Is. (Malvinas)	FLK	Falkland Is. Pound	FKP
Fiji	FJI	Fiji Dollar	FJD
Finland	FIN	Euro	EUR
France	FRA	Euro	EUR
France Metropolitan	FXX	Euro	EUR
French Guiana	GUF	Euro	EUR
French Polynesia	PYF	CFP Franc	XPF
French Southern Territory	ATF	Euro	EUR
Gabon	GAB	CFA Franc BEAC	XAF
Gambia	GMB	Dalasi	GMD
Georgia	GEO	Georgian Lari	GEL
Germany	DEU	Euro	EUR
Ghana	GHA	Cedi	GHC
Gibraltar	GIB	Gibraltar Pound	GIP

Greece	GRC	Euro	EUR
Greenland	GRL	Danish Krone	DKK
Grenada	GRD	E. Caribbean Dollar	XCD
Guadeloupe	GLP	Euro	EUR
Guam	GUM	U.S. Dollar	USD
Guatemala	GTM	Quetzal	GTQ
Guernsey	GGY	Pound Sterling	GBP
Guinea	GIN	Guinea Franc	GNF
Guinea—Bissau	GNB	Guinea-Bissau Peso	GWP
Guyana	GUY	Guyana Dollar	GYD
Haiti	HTI	Gourde	HTG
Heard and McDonald Is.	HMD	Australian Dollar	AUD
Holy See (Vatican City State)	VAT	Euro	EUR
Honduras	HND	Lempira	HNL
Hong Kong China	HKG	Hong Kong Dollar	HKD
Hungary	HUN	Forint	HUF
Iceland	ISL	Iceland Krona	ISK
India	IND	Indian Rupee	INR
Indonesia	IDN	Rupiah	IDR
Iran Airlines	--	Iranian Airline Rate	IRA
Iran Islamic Republic of	IRN	Iranian Rial	IRR
Iraq	IRQ	Iraqi Dinar	IQD
Ireland Republic of	IRL	Euro	EUR
Israel	ISR	New Israeli Shekel	ILS
Isle of Man	IMN	Pound Sterling	GBP
Italy	ITA	Euro	EUR
Jamaica	JAM	Jamaican Dollar	JMD
Jersey	JEY	Pound Sterling	GBP
Japan	JPN	Yen	JPY
Jordan	JOR	Jordanian Dinar	JOD
Kazakhstan	KAZ	Tenge	KZT
Kenya	KEN	Kenyan Shilling	KES
Kiribati	KIR	Australian Dollar	AUD
Korea Democratic People's Republic of (North Korea)	PRK	North Korean Won	KPW
Korea Republic of	KOR	Won	KRW
Kosovo	XKX	Euro	EUR
Kuwait	KWT	Kuwaiti Dinar	KWD

Kyrgyzstan	KGZ	Som	KGS
Lao People's Democratic Republic	LAO	Kip	LAK
Latvia	LVA	Latvian Lats	LVL
Lebanon	LBN	Lebanese Pound	LBP
Lesotho	LSO	Rand	ZAR
Liberia	LBR	Liberian Dollar	LRD
Libyan Arab Jamahiriya	LBY	Libyan Dinar	LYD
Liechtenstein	LIE	Swiss Franc	CHF
Lithuania	LTU	Lithuanian Litas	LTL
Luxembourg	LUX	Euro	EUR
Macau China	MAC	Pataca	MOP
Macedonia the Former Yugoslav Republic of	MKD	Denar	MKD
Madagascar	MDG	Malagasy Franc	MGF
Malawi	MWI	Malawi Kwacha	MWK
Malaysia	MYS	Malaysian Ringgit	MYR
Maldives	MDV	Rufiyaa	MVR
Mali	MLI	CFA Franc BCEAO	XOF
Malta	MLT	Maltese Lira	MTL
Marshall Islands	MHL	U.S. Dollar	USD
Martinique	MTQ	Euro	EUR
Mauritania	MRT	Ouguiya	MRO
Mauritius	MUS	Mauritius Rupee	MUR
Mayotte	MYT	Euro	EUR
Mexico	MEX	Mexican Peso	MXN
Micronesia	FSM	U.S. Dollar	USD
Moldova Republic of	MDA	Moldovan Leu	MDL
Monaco	MCO	Euro	EUR
Mongolia	MNG	Tugrik	MNT
Montenegro		Yugoslavian New Dinar	YUM
Montserrat	MSR	E. Caribbean Dollar	XCD
Morocco	MAR	Moroccan Dirham	MAD
Mozambique	MOZ	Metical	MZM
Myanmar	MMR	Kyat	MMK
Namibia	NAM	Namibia Dollar	NAD
Nauru	NRU	Australian Dollar	AUD
Nepal	NPL	Nepalese Rupee	NPR
Netherlands	NLD	Euro	EUR
Netherlands Antilles	ANT	Nether. Antillian Guilder	ANG

New Caledonia	NCL	CFP Franc	XPF
New Zealand	NZL	New Zealand Dollar	NZD
Nicaragua	NIC	Cordoba Oro	NIO
Niger	NER	CFA Franc BCEAO	XOF
Nigeria	NGA	Naira	NGN
Niue	NIU	New Zealand Dollar	NZD
Norfolk Is.	NFK	Australian Dollar	AUD
Northern Mariana Islands	MNP	U.S. Dollar	USD
Norway	NOR	Norwegian Krone	NOK
Oman	OMN	Rial Omani	OMR
Pakistan	PAK	Pakistan Rupee	PKR
Palau	PLW	U.S. Dollar	USD
Panama	PAN	Balboa	PAB
Papua New Guinea	PNG	Kina	PGK
Paraguay	PRY	Guarani	PYG
Peru	PER	Nuevo Sol	PEN
Philippines	PHL	Philippine Peso	PHP
Pitcairn	PCN	New Zealand Dollar	NZD
Poland	POL	Polish New Zloty	PLN
Portugal	PRT	Euro	EUR
Puerto Rico	PRI	U.S. Dollar	USD
Qatar	QAT	Qatari Rial	QAR
Reunion	REU	Euro	EUR
Romania	ROM	Leu	ROL
Russian Federation	RUS	Russian Ruble (International)	RUB
Russian Ruble (Domestic)	RUS	Russian Ruble (Domestic)	RUR
Rwanda	RWA	Rwanda Franc	RWF
Samoa	WSM	Tala	WST
San Marino	SMR	Euro	EUR
Sao Tome and Principe	STP	Dobra	STD
Saudi Arabia	SAU	Saudi Riyal	SAR
Senegal	SEN	CFA Franc BCEAO	XOF
Serbia	SRB	Serbian Dinar	RSD
Seychelles	SYC	Seychelles Rupee	SCR
Sierra Leone	SLE	Leone	SLL
Singapore	SGP	Singapore Dollar	SGD
St. Maarten	SXM	Netherlands Antillean Guilder	ANG
Slovakia	SVK	Slovak Koruna	SKK

Slovenia	SVN	Tolar	SIT
So. Georgia and So. Sandwich Is.	SGS	Pound Sterling	GBP
Solomon Is.	SLB	Solomon Is. Dollar	SBD
Somalia	SOM	Somali Shilling	SOS
South Africa	ZAF	Rand	ZAR
South Sudan	SSD	South Sudanese Pound	SSP
Spain	ESP	Euro	EUR
Sri Lanka	LKA	Sri Lanka Rupee	LKR
St. Helena	SHN	St. Helena Pound	SHP
St. Kitts-Nevis	KNA	E. Caribbean Dollar	XCD
St. Lucia	LCA	E. Caribbean Dollar	XCD
St. Pierre and Miquelon	SPM	Euro	EUR
St. Vincent and The Grenadines	VCT	E. Caribbean Dollar	XCD
Sudan	SDN	Sudanese Pound	SDD
Sudan Airlines	--	Sudan Airline Rate	SDA
Suriname	SUR	Surinam Guilder	SRG
Svalbard and Jan Mayen Is.	SJM	Norwegian Krone	NOK
Swaziland	SWZ	Lilangeni	SZL
Sweden	SWE	Swedish Krona	SEK
Switzerland	CHE	Swiss Franc	CHF
Syrian Arab Rep.	SYR	Syrian Pound	SYR
Taiwan	TWN	New Taiwan Dollar	TWD
Tajikistan	TJK	Somoni	TJS
Tanzania United Republic of	TZA	Tanzanian Shilling	TZS
Thailand	THA	Thailand Baht	THB
Togo	TGO	CFA Franc BCEAO	XOF
Tokelau	TKL	New Zealand Dollar	NZD
Tonga	TON	Pa'anga	TOP
Trinidad and Tobago	TTO	Trinidad and Tobago Dollar	TTD
Tunisia	TUN	Tunisian Dinar	TND
Turkey	TUR	Turkish Lira	TRL
Turkmenistan	TKM	Manat	TMM
Turks and Caicos Is.	TCA	U.S. Dollar	USD
Tuvalu	TUV	Australian Dollar	AUD
U.S. Minor Outlying Islands	UMI	U.S. Dollar	USD
U.S. Virgin Is.	VIR	U.S. Dollar	USD
Uganda	UGA	Uganda Shilling	UGX
Ukraine	UKR	Ukrainian Hryvnia	UAH

United Arab Emirates	ARE	U.A.E. Dirham	AED
United Kingdom	GBR	Pound Sterling	GBP
United States	USA	U.S. Dollar	USD
Uruguay	URY	Peso Uruguayo	UYU
Uzbekistan	UZB	Uzbekistan Sum	UZS
Vanuatu	VUT	Vatu	VUV
Venezuela	VEN	Bolivar	VEB
Vietnam	VNM	Dong	VND
Wallis and Futuna Is.	WLF	CFP Franc	XPF
Western Sahara	ESH	Moroccan Dirham	MAD
Yemen	YEM	Yemeni Rial	YER
Yugoslavia	YUG	Yugoslavian New Dinar	YUM
Zambia	ZMB	Zambian Kwacha	ZMK
Zimbabwe	ZWE	Zimbabwe Dollar	ZWD