MIB Payment Gateway

Developer Guide

Version 2.0

PAYMENT GATEWAY DEVELOPER GUIDE			
Prepared by:	Faseel Saeed	**DOC No:**	DG/20/01
Department:	Technology	**Approved by:**	Mohamed Ismath

DOCUMENT REVISION HISTORY			
Version	**Author**	**Change History**	**Date**
1.0	Faseel Saeed	Version 1	8-Jan-20

Contents

Overview

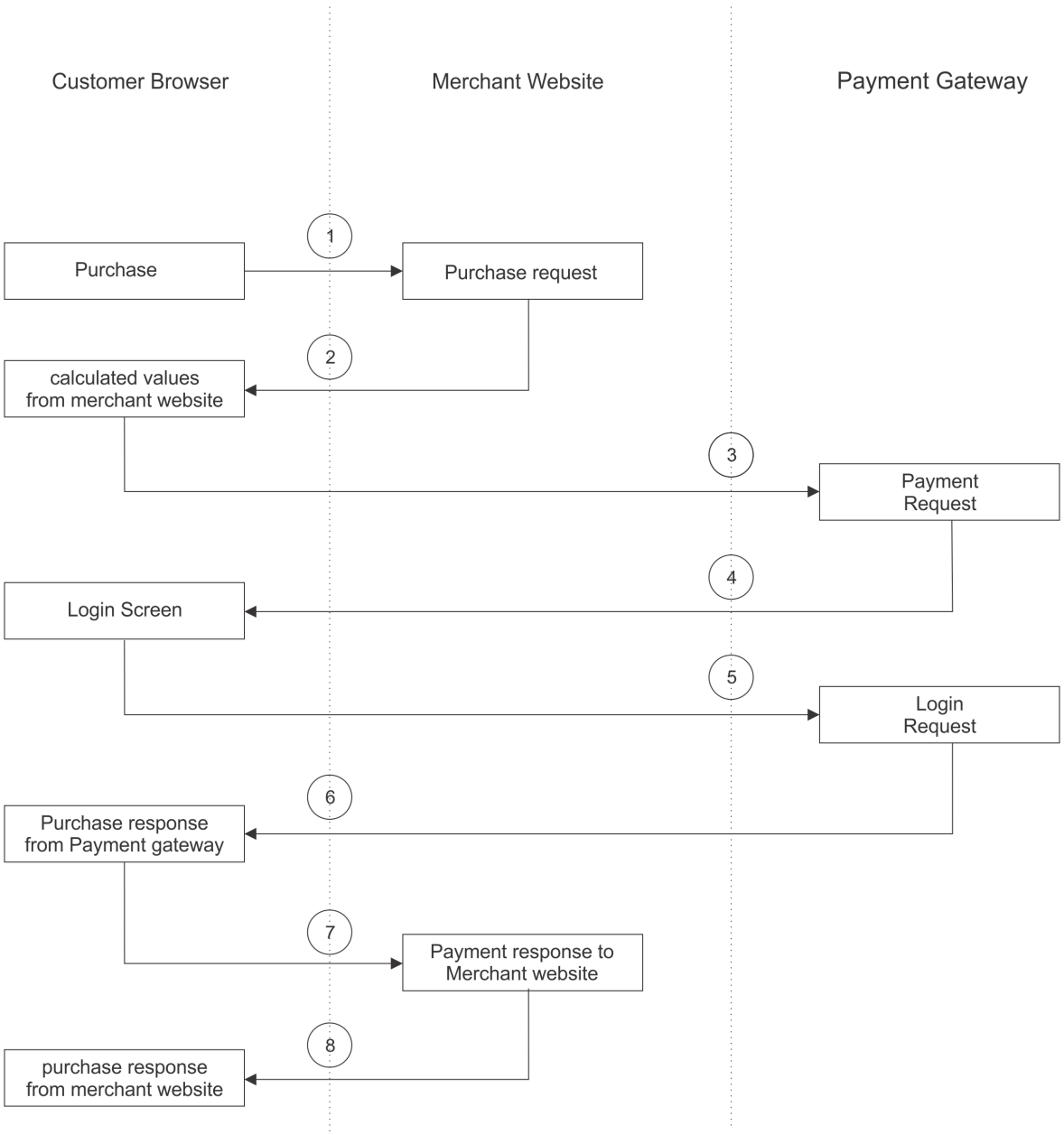

FIGURE 1. Payment process flow

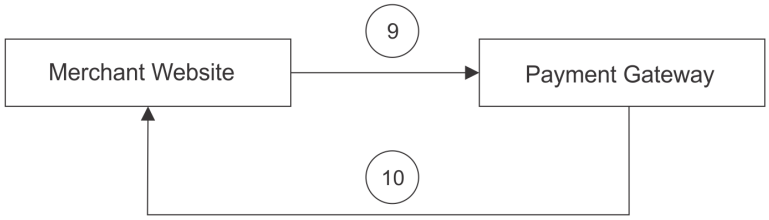

FIGURE 2. Reversal Request

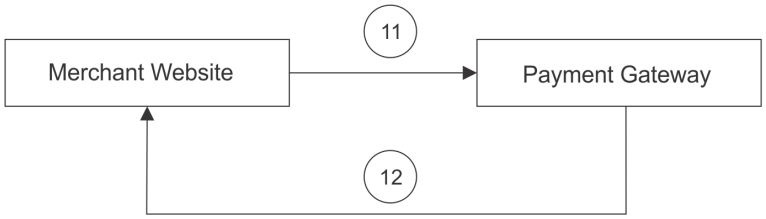

FIGURE 3. Status Request

1. The Customer visits the Merchant's website, retrieves the payment item (for example a bill) and selects to make payment by MIB Payment Gateway.
2. Merchant backend calculates the signature, sends it along with purchase amount and other information back to the customers browser with payment gateway redirect information
3. Users browser redirects to Payment gateway with pay information. Refer to *Message 1 – Payment Request* for details of the message.
4. The Payment Gateway verifies the Merchant's details and the integrity of the message received by verifying the hash signature. Refer to *Attachment 1 – Data Integrity Using Hash Signature* for details of the hash signature and Refer to *Attachment 2 – Sample Code to Test Hash Signature* for testing hash signature. The Payment Gateway displays the login screen for the customer to login.
5. Customer logs in to Payment gateway with Faisanet credentials
6. The Payment Gateway sends response message to customer's browser with redirect information to merchant's response URL. Refer to *Message 2 – Response from the Payment Gateway* for details of the message.
7. Users browser redirects to Merchants website and submits the Payment Gateway message values to the merchant website.
8. Merchant website updates its systems and shows a success or failure message. Merchant website should evaluate the message integrity by checking the signature.
9. If response (message 7) is not received, the Merchant's website can optionally initiate a reversal request (a kind of time-out reversal). Refer to *Message 3 – Reversal Request from Merchant's Website* for details of the message.
10. The Payment Gateway will reverse the transaction and send a reply. Refer to *Message 4 – Reversal Response from the Payment Gateway* for details of the message.

11. Merchant's website sends a status request to the Payment Gateway. Refer to *Message 5 – Transaction Status Request from Merchant's Website* for details of the message.
12. The Payment Gateway replies to the Merchant's website with a response message. Refer to *Message 6 – Transaction Status Response from the Payment Gateway* for details of the message.

MESSAGE 1 – Payment Request from Merchant Website

NAME	FORMAT	VALUE
version	AN(5)	This is the version of FaisaPay. It must be set to 2.
merID	N(15)	Merchant identification number provided by MIB.
acqID	N(11)	Acquirer ID provided by MIB.
merRespURL	AN(50)	The response URL for getting the result back. This is the URL of a web page on Merchant's server where response will be sent. The URL shall be provided to the bank for authorization.
purchaseCurrency	N(3)	The currency value of the purchase in ISO Numeric value. It will be either 462 or 840.
purchaseCurrencyExponent	N(1)	The number of decimal digits of the purchase currency. Usually 2.
orderID	AN(150)	The transaction ID of the order that uniquely identifies the transaction in the Merchant's system. It should be unique for the Merchant.
signatureMethod	AN(6)	Signature method used to calculate the signature (explained below). Supported Methods are: MD5, SHA1, SHA256. Default value if not present is SHA1.
purchaseAmt	N(12)	This is the total amount of the purchase. Purchase Amount as calculated by the Merchant's system.
signature	AN(56)	This is a digital signature that will verify that the contents of the request is not altered in transit. A hash signature calculated from the following: 1. password (This is a password provided by the Bank) 2. merID 3. acqID 4. orderID 5. purchaseAmt 6. purchaseCurrency 7. purchaseCurrencyExponent Signature Calculation: Base64 (HASH(password + merID + acqID + orderID + purchaseAmt + purchaseCurrency + purchaseCurrencyExponent)) The hash value that is produced using either MD5, SHA1 or SHA256 hash algorithm.

Note: *The purchaseAmt is straight forward, it is the payment amount, for example 156.97, but without the decimal place. Therefore, the amount 156.97 should be 15697.*

MESSAGE 2 – Response from Payment Gateway

NAME	FORMAT	VALUE
merID	N(15)	Merchant identification number provided by MIB.
acqID	N(11)	Acquirer ID provided by MIB.
orderID	AN(150)	The transaction ID of the order that uniquely identifies the transaction in the Merchant's system. It should be unique for the Merchant.
responseCode	N(1)	Indicates the result of the transaction. The result can be either an approval, decline, or error. Refer to *Message 3 – Response Codes* for more details.
reasonCode	N(3)	This is a code that can give more information to the Merchant about the transaction, such as what particular error occurred. Refer to *Message 4 – Response Reason Codes* for more details.
reasonCodeDesc	AN(100)	This is a text string that will briefly explain the type of response encountered.
referenceNo	AN(12)	A reference number that uniquely identifies the authorization in the Payment Gateway.
paddedCardNo	AN(15)	This value will only be present if the transaction was successful. It will contain a 6 digit that can identify the paying customer, used padded with leading "X" characters.
authCode	AN(6)	A valid Authorization Code in case the transaction was approved. For failed transactions, this value may not be present or maybe zero.
salt	AN(8)	An 8 digit random alpha numeric string.
signature	AN(56)	The hash value that is produced using either MD5, SHA1 or SHA256 hash algorithm on the following fields: 1. requestType – For Purchase request, this value is always 0 2. password (This is a password provided by the Bank) 3. merID 4. acqID 5. orderID 6. salt Signature Calculation: Base64 (HASH (requestType + password + merchant ID + acquirer ID + order ID + salt)) If the authorization request is successful Payment Gateway will create a second hashed signature and include it in the response

		message. Therefore when the Merchant receive the response data Merchant can verify that the values were not modified in transit. In case a response is in error, or declined, the Signature will not be present.
signatureMethod	AN(6)	Either MD5, SHA1 or SHA256 depending on the value sent by the merchant. Default value if not present is SHA1

MESSAGE 3 – Reversal request from merchant website.

NAME	FORMAT	VALUE
responseFormat	AN(5)	This is the return response format. It can be XML or JSON. Default if not present is XML
version	AN(5)	This is the version of FaisaPay. It has to be set to 2.
merID	N(15)	Merchant identification number provided by MIB.
acqID	N(11)	Acquirer ID provided by MIB.
requestType	N(1)	This is the request type of the transaction. For reversals, this value is 2.
orderID	AN(150)	The transaction ID of the order that uniquely identifies the transaction in the Merchant's system. It should be unique for the Merchant.
signatureMethod	AN(6)	Either MD5, SHA1 or SHA256 is the valid value. Default value if not present is SHA1.
signature	AN(56)	A hash signature calculated from the following: 1. password (This is a password provided by the Bank) 2. merID 3. acqID 4. orderID 5. requestType The hash value that is produced using either MD5, SHA1 or SHA256 hash algorithm Signature Calculation: Base64 (HASH(password + merID + acqID + orderID + requestType))

MESSAGE 4 – Reversal response from payment gateway

4.1 – XML format

The reversal response is passed in XML format. The following is a sample of a reversal response.

```
<response>
      <responseCode>1</responseCode>
      <reasonText> Request is approved.</reasonText>
      <reasonCode>108</reasonCode>
      <merID>1</merID>
      <acqID>25877145</acqID>
      <orderID>1010929988</orderID>
      <referenceNo/>
      <salt>xTvSuXlZ</salt>
      <signature>rMQCZKP2YxFQIeBQECyshBF2R9A=</signature>
      <signatureMethod>SHA1</signatureMethod>
</response>
```

4.2 – JSON format

The reversal response is passed in JSON format. The following is a sample of a reversal response.

```
{
  "responseCode": "1",
  "reasonText": " Request is approved.",
  "reasonCode": "108",
  "merID": "1",
  "acqID": "25877145",
  "orderID": "1010929988",
  "referenceNo": null,
  "salt": "pmDXNAmp",
  "signature": "IwodnlbMlgCaBI/xt5H5W+mLKrQ=",
  "signatureMethod": "SHA1"
}
```

4.3 – Response Message Description

NAME	FORMAT	VALUE
merID	N(15)	Merchant identification number provided by MIB.
acqID	N(11)	Acquirer ID provided by MIB.
requestType	N(1)	This is the request type of the transaction. For reversal request, this value is 2. This field may or may not be present in the response. However, this value will be used in signature calculation.

orderID	AN(150)	The transaction ID of the order that uniquely identifies the transaction in the Merchant's system. It should be unique for the Merchant.
salt	AN(8)	A random 8-digit alphanumeric value.
signatureMethod	AN(6)	Either MD5, SHA1 or SHA256 is the valid value. Default value if not present is SHA1.
signature	AN(56)	A hash signature calculated from the following: 1. password (This is a password provided by the Bank) 2. merID 3. acqID 4. orderID 5. requestType The hash value that is produced using either MD5, SHA1 or SHA256 hash algorithm Signature Calculation: Base64 (HASH(requestType + password + merID + acqID + orderID + salt)) **Note: Notice that the response signature calculation from the payment gateway is different from the request signature calculation.**

MESSAGE 5 – Status Request from Merchant Website

NAME	FORMAT	VALUE
responseFormat	AN(5)	This is the return response format. It can be XML or JSON. Default if not present is XML
version	AN(5)	This is the version of FaisaPay. It must be set to 2.
merID	N(15)	Merchant identification number provided by MIB.
acqID	N(11)	Acquirer ID provided by MIB.
requestType	N(1)	This is the request type of the transaction. For status request, this value is 1.
orderID	AN(150)	The transaction ID of the order that uniquely identifies the transaction in the Merchant's system. It should be unique for the Merchant.
signatureMethod	AN(6)	Either MD5, SHA1 or SHA256 is the valid value. Default value if not present is SHA1.
signature	AN(56)	A hash signature calculated from the following: 1. password (This is a password provided by the Bank) 2. merID 3. acqID 4. orderID 5. requestType The hash value that is produced using either MD5, SHA1 or SHA256 hash algorithm Signature Calculation: Base64 (HASH(password + merID + acqID + orderID + requestType))

MESSAGE 6 – Status response from Payment Gateway

6.1 – XML format

The status response is passed in XML format. The following is a sample of a status response.

```
<response>
      <success>1</success>
      <responseCode>1</responseCode>
      <reasonText>Request is successful</reasonText>
      <reasonCode>107</reasonCode>
      <requests>
            <trxRequest>
                  <requestTime>2019/12/16 08:01:07</requestTime>
                  <responseTime>2019/12/16 08:01:07</responseTime>
                  <responseCode>3</responseCode>
                  <reasonCode>36</reasonCode>
                  <reasonCodeDesc>Account holder cancelled the
request.</reasonCodeDesc>
                  <referenceNo>15497</referenceNo>
                  <purchaseCurrency>462</purchaseCurrency>
                  <evaluatedAmount>1</evaluatedAmount>
                  <authCode/>
                  <requestType>PAYMENT</requestType>
            </trxRequest>
            <trxRequest>
                  <requestTime>2019/12/16 08:02:07</requestTime>
                  <responseTime>2019/12/16 08:02:33</responseTime>
                  <responseCode>1</responseCode>
                  <reasonCode>1</reasonCode>
                  <reasonCodeDesc>Transaction is approved. </reasonCodeDesc>
                  <referenceNo>15499</referenceNo>
                  <purchaseCurrency>462</purchaseCurrency>
                  <evaluatedAmount>1</evaluatedAmount>
                  <authCode>2124072</authCode>
                  <requestType>PAYMENT</requestType>
            </trxRequest>
            <trxRequest>
                  <requestTime>2019/12/16 08:02:48</requestTime>
                  <responseTime>2019/12/16 08:02:48</responseTime>
                  <responseCode>2</responseCode>
                  <reasonCode>40</reasonCode>
                  <reasonCodeDesc>Duplicate Order Not Allowed</reasonCodeDesc>
                  <referenceNo>15500</referenceNo>
                  <purchaseCurrency>462</purchaseCurrency>
                  <evaluatedAmount>1</evaluatedAmount>
                  <authCode/>
                  <requestType>PAYMENT</requestType>
            </trxRequest>
            <trxRequest>
                  <requestTime>2019/12/16 08:03:29</requestTime>
                  <responseTime>2019/12/16 08:03:30</responseTime>
                  <responseCode>1</responseCode>
                  <reasonCode>1</reasonCode>
                  <reasonCodeDesc>Transaction is approved. </reasonCodeDesc>
                  <referenceNo>15501</referenceNo>
                  <purchaseCurrency>462</purchaseCurrency>
                  <evaluatedAmount>1</evaluatedAmount>
                  <authCode/>
```

```
                    <requestType>REVERSE</requestType>
                </trxRequest>
        </requests>
        <merID>1</merID>
        <acqID>25877145</acqID>
        <orderID>1010929988</orderID>
        <salt>yFsaBRKl</salt>
        <signature>1eH7fiwo7jqreTobRrFno29klZM=</signature>
        <signatureMethod>SHA1</signatureMethod>
</response>
```

6.2 – JSON format

```
{
  "success": true,
  "responseCode": "1",
  "reasonText": "Request is successful",
  "reasonCode": "107",
  "requests": [
    {
      "requestTime": "2019/12/16 08:01:07",
      "responseTime": "2019/12/16 08:01:07",
      "responseCode": "3",
      "reasonCode": "36",
      "reasonCodeDesc": "Account holder cancelled the request.",
      "referenceNo": "15497",
      "purchaseCurrency": "462",
      "evaluatedAmount": "1",
      "authCode": null,
      "requestType": "PAYMENT"
    },
    {
      "requestTime": "2019/12/16 08:02:07",
      "responseTime": "2019/12/16 08:02:33",
      "responseCode": "1",
      "reasonCode": "1",
      "reasonCodeDesc": "Transaction is approved.",
      "referenceNo": "15499",
      "purchaseCurrency": "462",
      "evaluatedAmount": "1",
      "authCode": "2124072",
      "requestType": "PAYMENT"
    },
    {
      "requestTime": "2019/12/16 08:02:48",
      "responseTime": "2019/12/16 08:02:48",
      "responseCode": "2",
      "reasonCode": "40",
      "reasonCodeDesc": "Duplicate Order Not Allowed",
      "referenceNo": "15500",
      "purchaseCurrency": "462",
      "evaluatedAmount": "1",
      "authCode": null,
      "requestType": "PAYMENT"
    },
    {
      "requestTime": "2019/12/16 08:03:29",
      "responseTime": "2019/12/16 08:03:30",
      "responseCode": "1",
      "reasonCode": "1",
      "reasonCodeDesc": "Transaction is approved.",
      "referenceNo": "15501",
```

```
      "purchaseCurrency": "462",
      "evaluatedAmount": "1",
      "authCode": null,
      "requestType": "REVERSE"
    }
  ],
  "merID": "1",
  "acqID": "25877145",
  "orderID": "1010929988",
  "salt": "cjWSal1k",
  "signature": "QiBvUEOaETAHxTcE5yw1jtqfc1U=",
  "signatureMethod": "SHA1"
}
```


6.3 – Status Response Message Description

NAME	FORMAT	VALUE
merID	N(15)	Merchant identification number provided by MIB.
acqID	N(11)	Acquirer ID provided by MIB.
requestType	N(1)	This is the request type of the transaction. For status request, this value is 1. This field may or may not be present in the response. However, this value will be used in signature calculation.
orderID	AN(150)	The transaction ID of the order that uniquely identifies the transaction in the Merchant's system. It should be unique for the Merchant.
salt	AN(8)	A random 8-digit alphanumeric value.
signatureMethod	AN(6)	Either MD5, SHA1 or SHA256 is the valid value. Default value if not present is SHA1.
signature	AN(56)	A hash signature calculated from the following: 1. password (This is a password provided by the Bank) 2. merID 3. acqID 4. orderID 5. requestType The hash value that is produced using either MD5, SHA1 or SHA256 hash algorithm Signature Calculation: Base64 (HASH (requestType + password + merID + acqID + orderID + salt)) **Note: Notice that the response signature calculation from the payment gateway is different from the request signature calculation.**

MESSAGE 7 – Response Codes

RESPONSE CODE	DESCRIPTION	DATA SENT WITH THE RESPONSE CODE
1	Approved	merID acqID orderID responseCode reasonCode reasonCodeDesc salt signature signatureMethod **Additional fields for Pay request** referenceNo paddedCardNo authCode
2	Declined	merID acqID orderID responseCode reasonCode reasonCodeDesc
3	Error	merID acqID orderID responseCode reasonCode reasonCodeDesc

MESSAGE 8 – Response Reason Codes

REASON CODE	REASON TEXT	NOTE	RESPONSE CODE
1	Transaction is approved.	A successful transaction.	1
2	Transaction is declined.	The customer does not have enough balance in his account to make the payment; amount is greater than account balance.	2
3	Transaction is declined.	Customer account/CIF status is not active (dormant, blacklisted, legal issue, etc).	2

4	Transaction is declined.	Merchant account/CIF status is not active (dormant, blacklisted, legal issue, etc).	2
5	Connection not secured.	Connection was not secured (HTTPS was not used).	3
6	HTTP Method not POST	HTTP Method not POST.	3
7	Field is missing.	A required field is missing in the order/transaction information sent by the Merchant.	3
8	Field format is invalid.	The order information sent by Merchant has a field with wrong data format or length. For example, (1) Text on purchaseAmt field. (2) acqID value is longer than 11. (3) Invalid merRespURL Refer to field types on respective request parameters.	3
10	Invalid Merchant	The Merchant does not exist.	2
11	Authentication Failed (Signature computed incorrectly).	Merchant was found but computed signature does not match the one included in the request.	2
12	Merchant is inactive.	Merchant status is inactive.	2
14	Merchant is not allowed to process this currency.	The Merchant is not allowed to process payments with the respective currency. For example, transactions in USD.	2
15	Merchant settings are not valid.	Merchant record is pending for approval or Merchant request came from a different URL than what is parameterized in the system.	2
17	Unable to process transaction.	The system could not debit the customer account or credit the Merchant account. The transaction failed from CBS for any other reason not stated here.	2
19	Unable to process transaction.	System cannot communicate to the database server.	3
32	Authentication failed.	The customer entered an invalid Username, Password or OTP.	2
35	Unable to process your request. Please try later.	Merchant exceeds allowed transaction amount or limit.	2
36	Account holder cancelled the request.	Customer clicks the Cancel button on payment screen.	2
38	Transaction processing terminated. Please try again later.	Transaction type, for example a reversal transaction, is not allowed for the Merchant.	2
40	Duplicate Order Not Allowed	Merchant sent an order with the OrderID same as one that had been sent previously, hence the new transaction is considered a duplicate. orderID must be unique for all successful transactions. If a previous request with the same orderID was unsuccessful, merchant can retry with the same orderID.	2

41	Account Holder Session Expired.	The customer was inactive for the allowed session time on the login or payment screen.	3
42	Illegal Operation by Account holder. Check Order Status.	The customer pressed the Back button on browser while the transaction was processing. Check the status of that order.	3
90	General Error during processing. Please try again later.	Any other error that is not specified here.	3
98	System is temporarily down. Try later.	System is temporarily down. For example, the application server is down or unreachable or the system is unable to connect to TCP port.	3
103	Order has being updated before.	A similar transaction was performed by the customer a short while ago.	3
104	Unable to update order at the moment. Try later.	System cannot update the order in database, for example, due to a unique key constraint.	3
105	Reversal time expired.	The time allowed to reverse the transaction has expired.	3
106	Invalid order.	Invalid order or order details not found.	3
107	Request is successful	Status request is successful.	1
108	Request is approved	Reversal Request is successful.	1
121	Order not found	Order not found in Payment Gateway system. This Occurs when a status request or reversal request is sent for an orderID.	2
122	Transaction is declined.	Customer daily limit reached	2
123	Transaction already reversed.	Transaction is already reversed in Payment Gateway system.	2
124	Transaction already reversed.	Transaction is already reversed in Payment Gateway system.	2
125	Order not found	Order not found in Payment Gateway system. This Occurs when a status request or reversal request is sent for an orderID.	2
126	Merchant does not have reversal rights.	Merchant does not have reversal rights.	2
127	Transaction is declined.	Zero amount transactions are declined.	2

ATTACHMENT 1 – Data Integrity using hash signature

The hash signature is a security feature that identifies the results of a transaction is from the appropriate Merchant and also for the Payment Gateway to make sure the integrity of data received on transaction request.

A unique signature or fingerprint of the transaction can be created using either MD5, SHA1 or SHA256 algorithm. This mathematical algorithm used to construct the signature is designed in such a way that any change to the information used in the calculation of the signature will cause a completely different signature to be created. In addition, the information used in the calculation of the signature cannot be discovered through any analysis of the signature itself; this is done by using information from the registered Merchant account. Every transaction that is processed through the system has a corresponding hash signature of the transaction created during the transaction process.

PAYMENT REQUEST FROM MERCHANT

The signature is included in the request and response of every payment transaction. The hash signature for a payment request is a hash of the following fields:

1. password (This is a password provided by the Bank)
2. merID
3. acqID
4. orderID
5. purchaseAmt
6. purchaseCurrency
7. purchaseCurrencyExponent

For example,

password	purple
merID	993123
acqID	25877145
orderID	521164321
purchaseAmt	12957
purchaseCurrency	462
purchaseCurrencyExponent	2

The hash would be run on the following string:

purple993123258771455211643211295746 22

The resulting hash signature value equals to (and encoded as a base 64 string, in this case):

Algorithm	Hash
SHA256	**kT/Lx+g/VQL7nmuPBpdGNCYeTpxMTdUVDJlHEfnX/PA=**
SHA1	**EwmMIjeVOWzCADBu9MHGhvdm3Pw=**
MD5	**eLd51CPRbwt7HDSuFaJxtw==**

PAYMENT RESPONSE FROM PAYMENT GATEWAY

The hash signature for a response to the above request is a hash of the following fields:

1. requestType – For Purchase request, this value is always 0
2. password (This is a password provided by the Bank)
3. merID
4. acqID
5. orderID
6. salt

For example,

requestType	0
password	purple
merID	993123
acqID	25877145
orderID	521164321
salt	Xvgb63hk

The hash would be run on the following string:

0purple993123258771455211643 21Xvgb63hk

The resulting hash signature value equals to (and encoded as a base 64 string):

Algorithm	Hash
SHA256	**VEy56a6tNuva0p9YO7O9fJU7eNleWUQc4TgpVIItKA0=**
SHA1	**LtkDujVM9KU0cZuhsVpgqeQaTcw=**
MD5	**qyhFs5V2HqaYLfDhMzv5ug==**

REVERSAL REQUEST FROM MERCHANT

The signature is included in the request and response of every reversal transaction. The hash signature for a reversal request is a hash of the following fields:

1. password (This is a password provided by the Bank)
2. merID
3. acqID
4. orderID
5. requestType

For example,

password	purple
merID	993123
acqID	25877145
orderID	521164321
requestType	2

The hash would be run on the following string:

purple99312325877145521164321 2

The resulting hash signature value equals to (and encoded as a base 64 string, in this case):

Algorithm	Hash
SHA256	**4ZxO0DAHF4yNxX65UjX+oJ8o6tsYZweReuYWTE4DmRg=**
SHA1	**0Fg0j9tUR0PCKIgew3qps/6TarA=**
MD5	**/uGWsfVYP4Y6bQHOm9QbKw==**

REVERSAL RESPONSE FROM PAYMENT GATEWAY

The hash signature for a response to the above request is a hash of the following fields:

1. requestType – For Purchase request, this value is always 0
2. password (This is a password provided by the Bank)
3. merID
4. acqID
5. orderID
6. salt

For example,

requestType	2
password	purple
merID	993123
acqID	25877145
orderID	521164321
salt	Xvgb63hk

The hash would be run on the following string:

2purple993123258771455211643 21Xvgb63hk

The resulting hash signature value equals to (and encoded as a base 64 string):

Algorithm	Hash
SHA256	**Au25HXPyBfbQlxB6uI3UvCfEPGIShD6hvptEM1b/pz0=**
SHA1	**9nreK7UAdBy1Xq6+CKr2CxVzEPY=**
MD5	**reL4fF3OsOKLGfn9ahKNUA==**

STATUS REQUEST FROM MERCHANT

The signature is included in the request and response of every status transaction. The hash signature for a reversal request is a hash of the following fields:

1. password (This is a password provided by the Bank)
2. merID
3. acqID
4. orderID
5. requestType

For example,

password	purple
merID	993123
acqID	25877145
orderID	521164321
requestType	1

The hash would be run on the following string:

purple9931232587714552116432 11

The resulting hash signature value equals to (and encoded as a base 64 string, in this case):

Algorithm	Hash
SHA256	**lUY6YcufeTJSoqiCku2rXCxZv6QlUh31uNsylJIK70I=**
SHA1	**TK31mk5LwW4OUYJt9+qE8X0e67k=**
MD5	**2X6y2crMqdEPKMuI7rs8uw==**

STATUS RESPONSE FROM PAYMENT GATEWAY

The hash signature for a response to the above request is a hash of the following fields:

1. requestType – For Purchase request, this value is always 0
2. password (This is a password provided by the Bank)
3. merID
4. acqID
5. orderID
6. salt

For example,

requestType	1
password	purple
merID	993123
acqID	25877145
orderID	521164321
salt	Xvgb63hk

The hash would be run on the following string:

1purple993123258771455211643 21Xvgb63hk

The resulting hash signature value equals to (and encoded as a base 64 string):

Algorithm	Hash
SHA256	**TWZoSAdgkO6H1zBtL5waL5NwKkOST26lKyM2B85q2Ew**
SHA1	**zm6LFUbTp/OFiw+H8ZobHaUZ2zE=**
MD5	**oonhYWRLRKLwLsL0IkHP0Q==**

ATTACHMENT 2 – SAMPLE CODE TO TEST HASH SIGNATURE

More information about the MD5, SHA1 and SHA256 hash algorithm, including sample implementation code, can be found in RFC1321, RFC3174, RFC6234 in The Internet Engineering Task Force web site respectively.

Using .NET you can use the available libraries to perform this:

```
SHA1 hashAlgorithm = SHA1.Create();
var encoder = new UTF8Encoding();
var bytes = encoder.GetBytes(strKey);

hashAlgorithm.ComputeHash(bytes);
var hash = Convert.ToBase64String(hashAlgorithm.Hash);

returnValue = hash;
```

Using PHP

```
<?php
$str = $_REQUEST['txt'];
echo '<br/> SHA1 => ';
echo hash('sha1',$str,false);
$s2=hash('sha1',$str,true);

echo '<br/>MD5 => ';
echo hash('md5',$str,false);
$s1=hash('md5',$str,true);

echo '<br/> MD5 + BASE64 => ';
echo base64_encode($s1);

echo '<br/> SHA1 + BASE64 => ';
echo base64_encode($s2);
?>
```