TerraPay Payment Gateway

Merchant Integration Kit Guidelines

1 Document History

Version	Submitted Date	Author	Revision History
0.1	06-05-2022	Girmiti Software	Initial draft
0.2	09-05-2022	Girmiti Software	Updated post feedback
0.3	30-05-2022	Girmiti Software	Updated document after including tokenization
0.4	08-06-2022	Girmiti Software	Updated document after tokenization query feedback
0.5	18-06-2022	Girmiti Software	Added postCardDetails API and iframe usage and code snippet
0.6	07-10-2022	Girmiti Software	Updated response samples and amount details.
0.7	28-02-2023	Girmiti Software	Added Status Check API Updated Error codes
0.8	29-03-2023	Girmiti Software	Updated explanation for request parameters of all api. Added explanation for embedded and redirect way of integration along with request and response samples.
0.9	23-06-2023	Girmiti Software	Added In-progress status under Transaction statuses in TerraPay PG section
1.0	30-11-2023	Girmiti Software	For order and billing data added State field for the transactions request where billing country is US and Canada.
1.1	13-12-2023	Girmiti Software	Added :processCheckout and postCarddetails,processTokenPayment. 1. authSuccessCallbackUrl 2. authDeclineCallbackUrl 3. authTimeoutCallbackUrl
1.2	27-03-2024	Girmiti Software	Added Browser Information and DeviceChannel, email id , cardHolderName and phone number in the request.

2 Introduction

TerraPay works with multiple customers and merchants to provide payment facilities across the globe.

TerraPay provides solution to support their customers, Merchants and Partners with easy interface to plugin to their existing Payment application to perform seamless payment transactions connecting to various connectors.

Further, to ensure the cardholder details are secured during the transaction and after, tokenization services are provided through TerraPay' s Payment Gateway ecosystem.

2.1 Purpose

TerraPay provides merchant seamless integration to Platform connecting to various connectors to perform Payment transactions.

Merchant Integration Kit provides technical integration point to TerraPay PG.

3 Merchant Integration

TerraPay PG provides an API called processCheckout which needs to be invoked by merchants who would like to integrate with TerraPay PG.

Note: If the Merchant wants to use TerraPay PG provided Card entry page, use the processCheckout API.

3.1 processCheckout API spec

Transaction	processCheckout
URL	https://<IP_Address>:8045/txn-service/processCheckout
Method	**POST**
Content Type	application/json

Note: A Sandbox URL will be provided for the Merchants to perform the testing and before going live.

3.1.1 Request Parameters

Parameter Name	Data Type	(M/O/C)	Description
merchantId	String	M	Alphanumeric string without special characters, 1-10 in length
remittanceId	String	C	Alphanumeric string received from remittance platform. 16 characters in length. For remittance transaction this has value has to be sent, for pure acquiring this value can be left empty.
address1	String	M	Alphanumeric string without special characters. If any will be URL encoded.
city	String	M	1-30 characters
state	String	C	For US and Canada, state has to be passed

Parameter Name	Data Type	(M/O/C)	Description
country	String	M	ISO country code, 1-3 characters
email	String	M	Alphanumeric string without special characters, except allowed email characters, 1-100 in length.
firstName	String	M	Alphanumeric string without special characters, 1-22 in length
lastName	String	M	Alphanumeric string without special characters, 1-22 in length
phoneNumber	String	M	Number without country code, 1-13 in length.
postalCode	String	M	Alphanumeric string without special characters, 1-13 in length
currency	String	M	3 letter ISO currency code
totalAmount	String	M	Amount in cents without decimals, 1-10 in length
embeded_system	String	C	"yes" if merchant wants to embed the card details page in his webpage. Else do not send this. "no" if merchant does not want to embed the card details page in his webpage.
partnerImtTxnRefno	String	M	Alphanumeric string. This is the merchant transaction reference id which merchant needs to generate and pass.
partnerImtTxnAuthCode	String	M	Alphanumeric string (Can send the same value as remittanceId)
paymentType	String	M	Value should be "card" for card payment. Value should be "token" when using saved token for payment.
authSuccessCallbackUrl	String	O	A valid URL has to be provided by the merchant, if merchant wishes to receive auth success response on a particular URL. If this field does not have a value, auth success response will be sent on the callback url provided when the merchant onboarding was done.
authDeclineCallbackUrl	String	O	A valid URL has to be provided by the merchant, if merchant wishes to receive auth failure response on a particular URL. If this field does not have a value, auth failure response will be sent on the

Parameter Name	Data Type	(M/O/C)	Description
			callback url provided when the merchant onboarding was done.
authTimeoutCallbackUrl	String	O	A valid URL has to be provided by the merchant, if merchant wishes to receive auth timeout response on a particular URL. If this field does not have a value, auth timeout response will be sent on the callback url provided when the merchant onboarding was done.
tokenNumber	String	C	token value should be sent if paymentType = “token”, else should be empty.
deviceChannel	String	C	As transaction provide web-view. Set Value of Device Channel as “Browser”.
browserInfo	**JSON Object**		
browserLanguage	String	O	Value represents the browser language as defined in IETF BCP47. Example:en-US
acceptHeader	String	O	Value of the Accept header sent by the client’s web browser.
screenWidth	String	O	Total width of the cardholder's screen in pixels. Example: 1536.
javaEnabled	String	O	A Boolean value that represents the ability of the cardholder browser to execute Java. Value is returned from the navigator.javaEnabled property. Possible Values:True/False
browserIpAddress	String	O	Browser IP Address
screenHeight	String	O	Total height of the Cardholder's scree in pixels, example: 864.
browserName	String	O	Value of the User-Agent header sent by the client’s web browser.
timeDifference	String	O	Time difference between UTC time and the cardholder browser local time, in minutes, Example:300
colorDepth	String	O	Value represents the bit depth of the color palette for displaying images, in bits per pixel.
contentType	String	O	content type of the HTTP accept header.
jsEnabled	String	O	A Boolean value that represents the ability of the cardholder browser to execute JavaScript. Possible Values:True/False. Note: Merchants should be able to know the values from fingerprint details of cardholder's browser.
browserEmail	String	O	Send the same value as email. Alphanumeric string without special

Parameter Name	Data Type	(M/O/C)	Description
			characters, except allowed email characters, 1-100 in length

3.1.2 Request Header

Parameter Name	Data Type	(M/O/C)	Description
authorization	String	O	SHA 256 hash value of the JSON request body; encrypted with a predefined key. Encryption algo to be used: AES/ECB/PKCS5Padding

* **M**- Mandatory; **O**- Optional; **C-** Conditional

Note:
The predefined key is shared to the merchant during merchant onboarding.

Merchants are required to post the above parameters to processCheckout API of TerraPay PG. The key issued will be an AES 256 key, the key issued will be base64 encoded string, which should be decoded first and then be used for encryption. Algorithm used to encrypt the card details should be AES/ECB/PKCS5Padding.

Sample request to be built by Merchant.

```
{"authorization":"qoi96Yv4FeAXUta6OOzCTICTvq7mE79edxwKGGTXFRfJ6H1\/s+X+dv350aljNx+XLMWCH3XkJZtjVxoy8Hy
qmWZaYamttq\/BP\/pcxovE8vk=","data":[{"lastName":"C","country":"US","city":"California","address1":"13
150
Abana","postalCode":"90703","partnerImtTxnRefno":"PARTNWEIMT345","deviceChannel":"Browser","embeded_sy
stem":"yes","payment_link":"yes","paymentType":"card","authDeclineCallbackUrl":"http://testPG.in/decli
ne","firstName":"Raghu","totalAmount":"1","authTimeOutCallbackUrl":"http://testPG.in/timeout","phoneNu
mber":"7829936352","remittanceId":"REMITTANCE34567","merchantId":"614967544947403","partnerImtTxnAuthC
ode":"1200","authSuccessCallbackUrl":"http://testPG.in/success","currency":"USD","state":"CA","email":
"raghu.chakali@girmiti.com","browserInfo":{"browserEmail":"raghu.chakali@girmiti.com","browserLanguage
":"en-
US","screenWidth":1920,"acceptHeader":"text/html,application/xhtml+xml,application/xml;q=0.9,image/avi
f,image/webp,image/apng,*/*;q=0.8,application/signed-
exchange;v=b3;q=0.7","javaEnabled":false,"screenHeight":839,"browserIpAddress":"49.204.88.98","browser
Name":"Google                                               Chrome","colorDepth":24,"timeDifference":-
330,"jsEnabled":true,"contentType":"application/x-www-form-urlencoded"}}]}
```

Then this whole request should be Base64 Url encoded and sent to PG via a form submit. Below is the sample Base64 url encoded data.

```
element=eyJhdXRob3JpemF0aW9uIjoicW9pOTZZdjRGZUFYVXRhNk9PekNUSUNUdnE3bUU3OWVkeHdLR0dUWEZSZko2S
DFcL3MrWCtkdjM1MGFsak54K1hMTVdDSDNYa0padGpWeG95OEh5cW1XWmFZYW10dHFcL0JQXC9wY3hvdkU4dms9Iiwi
ZGF0YSI6W3sibGFzdE5hbWUiOiJDIiwiY291bnRyeSI6IlVTIiwiY2l0eSI6IkNhbGlmb3JuaWEiLCJhZGRyZXNzMSI6IjEzMTUwIE
FiYW5hIiwicG9zdGFsQ29kZSI6IjkwNzAzIiwicGFydG5lckltdFR4blJlZm5vIjoiUEFSVE5XRUlNVDM0NSIsImRldmljZUNoYW5uZ
WwiOiJCcm93c2VyIiwiZW1iZWRlZF9zeXN0ZW0iOiJ5ZXMiLCJwYXltZW50X2xpbmsiOiJ5ZXMiLCJwYXltZW50VHlwZSI6ImNh
cmQiLCJhdXRoRGVjbGluZUNhbGxiYWNrVXJsIjoiaHR0cDovL3Rlc3RQRy5pbi9kZWNsaW5lIiwiZmlyc3ROYW1lIjoiUmFnaHUi
LCJ0b3RhbEFtb3VudCI6IjEiLCJhdXRoVGltZU91dENhbGxiYWNrVXJsIjoiaHR0cDovL3Rlc3RQRy5pbi90aW1lb3V0IiwicGhvbm
VOdW1iZXIiOiI3ODI5OTM2MzUyIiwicmVtaXR0YW5jZUlkIjoiUkVNSVRUQU5DRTM0NTY3IiwibWVyY2hhbnRJZCI6IjYxNDk2N
zU0NDk0NzQwMyIsInBhcnRuZXJJbXRUeG5BdXRoQ29kZSI6IjEyMDAiLCJhdXRoU3VjY2Vzc0NhbGxiYWNrVXJsIjoiaHR0cD
ovL3Rlc3RQRy5pbi9zdWNjZXNzIiwiY3VycmVuY3kiOiJVU0QiLCJzdGF0ZSI6IkNBIiwiZW1haWwiOiJyYWdodS5jaGFrYWxpQG
dpcm1pdGkuY29tIiwiYnJvd3NlckluZm8iOnsiYnJvd3NlckVtYWlsIjoicmFnaHUuY2hha2FsaUBnaXJtaXRpLmNvbSIsImJyb3dzZX
JMYW5ndWFnZSI6ImVuLVVTIiwic2NyZWVuV2lkdGgiOjE5MjAsImFjY2VwdEhlYWRlciI6InRleHQvaHRtbCxhcHBsaWNhdGlvbi
94aHRtbCt4bWwsYXBwbGljYXRpb24veG1sO3E9MC45LGltYWdlL2F2aWYsaW1hZ2Uvd2VicCxpbWFnZS9hcG5nLCovKjtxPT
AuOCxhcHBsaWNhdGlvbi9zaWduZWQtZXhjaGFuZ2U7dj1iMztxPTAuNyIsImphdmFFbmFibGVkIjpmYWxzZSwic2NyZWVuSG
VpZ2h0Ijo4MzksImJyb3dzZXJJcEFkZHJlc3MiOiI0OS4yMDQuODguOTgiLCJicm93c2VyTmFtZSI6Ikdvb2dsZSBDaHJvbWUiLC
```

Jjb2xvckRlcHRoIjoyNCwidGltZURpZmZlcmVuY2UiOi0zMzAsImpzRW5hYmxlZCI6dHJ1ZSwiY29udGVudFR5cGUiOiJhcHBsaWNhdGlvbi94LXd3dy1mb3JtLXVybGVuY29kZWQifX1dfQ

3.1.3 Recommended Approach

We would recommend merchant to post the billing and order data in encrypted format to an API in the merchant application. This API should in turn call the processCheckout API provided by TerraPay PG over https.

To invoke TerraPay PG API over https, TerraPay PG will provide SSL certificates for handshake.

There are 2 ways to get the card details page.

- Embedding the html content sent by PG.
- Redirecting to PG server.

3.1.3.1 Embedding

Merchant can embed the html content sent by processCheckout API onto his web page. This can be done by adding an iFrame element on his webpage and setting the html content received from PG to the iFrame.

Once the content is loaded in iFrame, user can enter the card details and submit the form, if challenge is required the OTP page is displayed to the user, once user enters OTP and submits the form, PG will check if authentication is successful, if authentication is successful, PG will initiate authorization, once PG gets the authorization response, PG will send back the response in the form of html to the merchant.

Response will be base64 encoded, merchant has to decode the data and fetch the details. Below is a sample response. The field transation_response will have the base64 encoded transaction response, which should be decoded by merchant to get the authorization details.

```
{
        "event_id":"event.completed",
        "message":"completed",
        "transaction_response":"ewogICJ0cmFuc2FjdGlvbklkIiA6ICIxMDAwMDAxMjU2NzEiLAogICJyZXNwb25zZUNvZGUiIDogIjIwMSIsCiAgInN0YXR1cyIgOiAiQVVUSE9SSVpFRCIsCiAgImF1dGhvcml6ZWRBbW91bnQiIDogIjEzMC43NiIsCiAgInRvdGFsQW1vdW50IiA6ICIxMzAuNzYiLAogICJjdXJyZW5jeSIgOiAiSVNLIiwKICAidHhuUmVmTnVtIiA6ICIzMDU5MTMyMzA2ODMiLAogICJyZW1pdHRhbmNlSWQiIDogIlRQQ1AwNDU3NTY3MjI2NiIsCiAgIm1lcmNoYW50SWQiIDogIjY1MDcxNTk1MjMwMTgxMiIsCiAgInRva2VuQ2FyZCIgOiB7CiAgICAibWFza2VkUEFOIiA6ICI0NzYxMzRYWFhYWFgwMTE4IgogIH0sCiAgInBhcnRuZXJJTVRUeG5SZWYiIDogIjEyMDAiLAogICJwYXJ0bmVySW10VHhuQXV0aENvZGUiIDogIjEyMDAiLAogICJwYXJ0bmVySW10VHhuUmVmbm8iIDogIjEyMDAiCn0=",
        "status":true
}
```

To receive the response sent by PG, merchant has to add the below listener to his page.

```
window.addEventListener("message", function(event) {
console.log('simple listener');
console.log(event.origin);
  if (event.origin === "url will be provided") {
    console.log(event.data);
  }
```

```
                }, false);
```

3.1.3.2 Re-direction method

Another way of using card details page provided by TerraPay PG is to do a form submit to TerraPay PG provided endpoint, TerraPay PG server will display the card details page and subsequently will handle the complete process of displaying the challenge flow page to the customer. Once the TerraPay PG receives the result of authorization, the same will be posted to merchant callback url and remittance callback url. Once the merchant receives the result of authorization, merchant can then decide and display appropriate message to the customer.

Merchant is required to provide a callback URL to TerraPay PG during onboarding. This callback URL will be invoked by TerraPay PG to inform merchant about the result of authorization request.

Note: Merchant who is using his own card details page. He should be invoking the postCardDetails API instead of above processCheckout API.

For redirection to work merchant has to add the authorization data in the request body as shown in the sample request, and has to do the form submit.

3.2 postCardDetails API

Below are the parameters that need to be sent by merchant when invoking API.

3.2.1 Request Parameters

Parameter Name	**Data Type**	**(M/O/C)**	**Description**
merchantId	String	M	Alphanumeric string without special characters, 1-10 in length
remittanceId	String	C	Alphanumeric string received from remittance platform. 16 characters in length. For remittance transaction this has value has to be sent, for pure acquiring this value can be left empty.
address1	String	M	Alphanumeric string without special characters. If any will be URL encoded.
city	String	M	1-30 characters
state	String	C	For US or Canada billing countries required state field to be passed for both CyberSource and Valitor Transactions.
country	String	M	ISO country code, 1-3 characters
email	String	M	Alphanumeric string without special characters, except allowed email characters, 1-100 in length

Parameter Name	Data Type	(M/O/C)	Description
firstName	String	M	Alphanumeric string without special characters, 1-22 in length
lastName	String	M	Alphanumeric string without special characters, 1-22 in length
phoneNumber	String	M	Number without country code, 1-13 in length
postalCode	String	M	Alphanumeric string without special characters, 1-13 in length
currency	String	M	3 letter ISO currency code
amount	String	M	Amount with 2 decimal places
partnerImtTxnRefno	String	M	Alphanumeric string. This is the merchant transaction reference id which merchant needs to generate and pass.
partnerImtTxnAuthCode	String	M	Alphanumeric string (Can send the same value as remittanceId)
CardNumber	String	M	Encrypted card number
CVV	String	M	Encrypted CVV
expiryMonth	String	M	Encrypted expiryMonth
expiryYear	String	M	Encrypted expiryYear
paymentType	String	M	Value should be "card"
saveCard	String	M	Value should be "true" if card has to be tokenized. Value should be "false" if card does not have to be tokenized.
authSuccessCallbackUrl	String	C	When the request contains authSuccessCallbackUrl parameter and if the transaction got authorized we will send transaction response in decode format to the authSuccessCallbackUrl url. If the request doesn't contains authSuccessCallbackUrl we will send the transaction response to onboarded merchant callback url.
authDeclineCallbackUrl	String	C	When the request contains authDeclineCallbackUrl parameter and if the transaction got authorized we will send transaction response in decode format to the authDeclineCallbackUrl url.

Parameter Name	Data Type	(M/O/C)	Description
			If the request doesn't contains authDeclineCallbackUrl we will send the transaction response to onboarded merchant callback url.
authTimeoutCallbackUrl	String	C	When the request contains authTimeoutCallbackUrl parameter and if the transaction got authorized we will send transaction response in decode format to the authTimeoutCallbackUrl url. If the request doesn't contains authTimeoutCallbackUrl we will send the transaction response to onboarded merchant callback url.
embeded_system	String	C	"yes" if merchant wants to embed the card details page in his webpage. Else do not send this. "no" if merchant does not want to embed the card details page in his webpage.
deviceChannel	String	C	As Transaction provide web-view. Device Channel will Value should be "Browser".
cardHolderName	String	M	While performing the transaction, it is required to pass the card holder name in encryted format.
browserInfo	**JSON Object**		
browserLanguage	String	O	Value represents the browser language as defined in IETF BCP47. Example:en-US
acceptHeader	String	O	Value of the Accept header sent by the client's web browser.
screenWidth	String	O	Total width of the cardholder's screen in pixels. Example: 1536.
javaEnabled	String	O	A Boolean value that represents the ability of the cardholder browser to execute Java. Value is returned from the navigator.javaEnabled property. Possible Values:True/False
browserIpAddress	String	O	Browser IP Address
screenHeight	String	O	Total height of the Cardholder's scree in pixels, example: 864.
browserName	String	O	Value of the User-Agent header sent by the client's web browser.
timeDifference	String	O	Time difference between UTC time and the cardholder browser local time, in minutes, Example:300

Parameter Name	Data Type	(M/O/C)	Description
colorDepth	String	O	Value represents the bit depth of the color palette for displaying images, in bits per pixel.
contentType	String	O	content type of the HTTP accept header.
jsEnabled	String	O	A Boolean value that represents the ability of the cardholder browser to execute JavaScript. Possible Values:True/False. Note: Merchants should be able to know the values from fingerprint details of cardholder's browser.
browserEmail	String	O	Send the same value as email. Alphanumeric string without special characters, except allowed email characters, 1-100 in length

3.2.2 Request Header

Parameter Name	Data Type	(M/O/C)	Description
authorization	String	M	SHA 256 hash value of the JSON request body; encrypted with a predefined key.

Merchant has to use the API key shared with them after onboarding, to encrypt the card details. The key issued will be an AES 256 key, the key issued will be base64 encoded string, which should be decoded first and then be used for encryption. Algorithm used to encrypt the card details should be AES/ECB/PKCS5Padding.

Terrapay PG will validate the request parameters, decrypt the card number, cvv, expiry month and expiry year with the API key that was shared with the merchant.

If challenge flow is demanded by the Issuer, postCardDetails API will return the form content which needs to be displayed by the merchant by displaying the form content in an iframe.

Below is the sample iframe snippet for reference

```
<div id="iframeDiv" style="display: none;" class="center" class="wrapper">
        <iframe id="iframeId" name="iframeName" height="400" width="400" frameBorder="0"></iframe>
</div>
```

Below is the code that can be used to load the form content into iframe.

```
document.getElementById(iframeId ').src = "data:text/html;charset=utf-8,"+ escape(data);
```

Below is an example of form content that will be sent for the merchant to load into an iframe.

```
<html class="no-js" lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
        <META http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <meta charset="utf-8">
    <title>3D Secure Processing</title>
    <link href="https://mpi.xyz.com/mdpaympi/static/mpi.css" rel="stylesheet" type="text/css">
</head>
<body>
    <div id="main">
        <div id="content">
            <div id="order">
                <h2>3D Secure Processing</h2>
                <script src="https://mpi.xyz.com/mdpaympi/static/red.js" defer>/* needed
for xsl to xhtml */</script>
                <div id="spinner">
                    <img src="https://mpi.xyz.com/mdpaympi/static/preloader.gif"
alt="Please wait.."></div>
                    <img src="https://mpi.
xyz.com/mdpaympi/static/mc_idcheck_hrz_ltd_pos_103px.png" alt="MasterCard ID Check">
                <div id="formdiv"><div>
                    <form id="webform0" name="ddcoll" method="POST"
action="https://mpi.xyz.com/mdpaympi/MerchantServer" accept_charset="UTF-8">
                        <input type="hidden" name="txid" value="213583741">
                        <input type="hidden" name="TDS2_Navigator_language"
value="">
                        <input type="hidden"
name="TDS2_Navigator_javaEnabled" value="">
                        <input type="hidden"
name="TDS2_Navigator_jsEnabled" value="">
                        <input type="hidden" name="TDS2_Screen_colorDepth"
value="">
                        <input type="hidden" name="TDS2_Screen_height"
value="">
                        <input type="hidden" name="TDS2_Screen_width"
value="">
                        <input type="hidden" name="TDS2_Screen_PixelDepth"
value="">
                        <input type="hidden" name="TDS2_TimezoneOffset"
value="">
                        <input type="hidden" name="digest"
value="BDTSh8vu5NYbpQcq2rs1cLJNiUrCtrFTMbIKkbO809Q=" readonly="true">
                        <input type="hidden" name="transientData"
value="E5DzjQ/VI3sHU+j8Ob8B5FckHPzMeYKsD0k3U+yBM5TEkW85XHOkN4Jx4MLTJHytb6PmG/mWpN3bF1qct
7PiGIynC8UN1eMywONkTq2/24d2LTXWOyghhoV/0wvjRK93/Na0MUHVHAgNgLMn1dHMytH+nZ4DvGc9QtpCajcP
yi2TyUt4wjz0aSGswuxEkNGl9g3xdcW4XAqtlAzSzd+SysKUv0xoL6szS2JTq+o7SBdsn+1uHkgCYO3uI6MBamfZN1
RQ9gvUHqudZ8lyUzjxfbvlAXEdX/lKWbxxX1r0bsnu4MF1Us+kFlb1GnCM41+FaWL98ue7l2G2WHmEa2JvqcrfWEdL
snpA7KUDfrhiBkb54sDQ/M8UaULhhdUFntt3TMgnKVIXIqMbyHm0RPOL8JOhb8RpwzJVz3d7BQuEHJEM+VJBRm2
MwQcb4XqYlRGlfeBPZCmPaEFb5KBJlbVZxYGK5T2ulq+ofkT+gqs0qkfNG4NuTwCl7nIR2YQG8iqnUR5KP+7BobPM
eckZXVsJ0do/+CCUC+MG99I0rOmbaveNANa0Fp+jc8d27RMX/1n1">
                        <input type="submit" name="submitBtn" id="submitBtn"
value="Please click here to continue">
                    </form>
                </div>
            </div>
            <noscript>
                <div align="center"><b>Javascript is turned off or not
supported!</b><br></div>
            </noscript>
        </div>
        <div id="content-footer"></div>
    </div>
</div>
</body>
</html>
```

3.3 Callback Response to Merchant application

TerraPay PG will send the response of transaction processing back to merchant application to the preconfigured call back url in below format.

Please note this response is in the same format from both processCheckout and postCardDetails API

Response Parameter	Description
status	Transaction status
paymentId	Network payment identifier
authorizedAmount	Amount authorized
totalAmount	Total amount
currency	3 letter ISO currency code
txnRefNum	TerraPay PG internal reference number
merchantId	TerraPay PG merchant ID
remittanceId	Incoming remittance ID
partnerImtTxnRefno	Incoming transaction ref Id
partnerImtTxnAuthCode	Incoming transaction auth Id
tokenCard	Tokenized card

Sample success response

```
{
  "transactionId" : "100000020767",
  "status" : "AUTHORIZED",
  "authorizedAmount" : "100",
  "totalAmount" : "100",
  "currency" : "ISK",
  "txnRefNum" : "227809789099",
  "remittanceId" : "QR037CK3KVGSYR4CE",
  "merchantId" : "650715952301812",
  "tokenCard" : {
    "tokenPAN" : "ce0f1459-bbf4-4c59-8fcb-9e72dc549162",
    "maskedPAN" : "476134XXXXXX0399"
  },
  "partnerImtTxnAuthCode" : "QR037CK3KVGSYR4CE",
  "partnerImtTxnRefno" : "ea1ff362-6e52-45f3-b235-9bf65f2730c3"
}
```

Tokenization

Introduction

So as to ensure that cardholder sensitive information, especially the PAN is secured during a payment transaction, the PAN is tokenized by TerraPay' s Payment Gateway ecosystem. This token, which is unique and mapped to each individual card the first time the card is used to perform a payment at an online merchant location, is subsequently used for secure transaction processing from then on.

TerraPay' s Payment Gateway provides the tokenization services for online merchants and the following section details how the tokenization and related services have to be invoked during the online payment transaction.

3.4 processTokenPayment API spec

Transaction	**Process Token Payment**
URL	https://<IP_Address>:8045/txn-service/eComm/processTokenPayment
Method	**POST**
Content Type	application/json

Note: A Sandbox URL will be provided for the Merchants to perform the testing and before going live.

processTokenPayment API is used to initiate a token-based transaction. The merchant needs to send the token number along with the details mentioned in the below tables.

3.4.1 Request Parameters

Parameter Name	**Data Type**	**(M/O/C)**	**Description**
merchantId	String	M	Alphanumeric string without special characters, 1-10 in length
remittanceId	String	C	Alphanumeric string received from remittance platform. 16 characters in length. For remittance transaction this has value has to be sent, for pure acquiring this value should be empty.
currency	String	M	3 letter ISO currency code
amount	String	M	Amount with 2 decimal places
virtualCard	String	M	Tokenized card that is created when the customer opts for tokenizing his card
tokenType	String	M	Parameter for signifying if it is a recurring, installment or single transaction.

Parameter Name	Data Type	(M/O/C)	Description
			Takes values of R (Recurring), I (Installment), C (Customer initiated), M (Merchant initiated) transaction.
partnerImtTxnRefno	String	M	Alphanumeric string. This is the merchant transaction reference id which merchant needs to generate and pass.
partnerImtTxnAuthCode	String	M	Alphanumeric string (Can send the same value as remittanceId)
state	String	C	For US or Canada state has to be provided.
authSuccessCallbackUrl	String	C	When the request contains authSuccessCallbackUrl parameter and if the transaction got authorized we will send transaction response in Base64 encoded format to the authSuccessCallbackUrl url. If the request doesn't contains authSuccessCallbackUrl we will send the transaction response to onboarded merchant callback url.
authDeclineCallbackUrl	String	C	When the request contains authDeclineCallbackUrl parameter and if the transaction got authorized we will send transaction response in Base64 encoded format to the authDeclineCallbackUrl url. If the request doesn't contains authDeclineCallbackUrl we will send the transaction response to onboarded merchant callback url.
authTimeoutCallbackUrl	String	C	When the request contains authTimeoutCallbackUrl parameter and if the transaction got authorized we will send transaction response in Base64 encoded format to the authTimeoutCallbackUrl url. If the request doesn't contains authTimeoutCallbackUrl we will send the transaction response to onboarded merchant callback url.
firstName	String	M	Alphanumeric string without special characters, 1-22 in length
lastName	String	M	Alphanumeric string without special characters, 1-22 in length
address1	String	M	Alphanumeric string without special characters. If any will be URL encoded.
city	String	M	1-30 characters

Parameter Name	**Data Type**	**(M/O/C)**	**Description**
country	String	M	ISO country code, 1-3 characters
postalCode	String	M	Alphanumeric string without special characters, 1-13 in length
email	String	M	Alphanumeric string without special characters, except allowed email characters, 1-100 in length
phoneNumber	String	M	Number without country code, 1-13 in length
embeded_system	String	C	"yes" if merchant wants to embed the card details page in his webpage. Else do not send this. "no" if merchant does not want to embed the card details page in his webpage.
deviceChannel	String	O	As Transaction provide web-view. Device Channel value should be "Browser".
browserInfo	**JSON Object**		
browserLanguage	String	O	Value represents the browser language as defined in IETF BCP47. Example:en-US
acceptHeader	String	O	Value of the Accept header sent by the client's web browser.
screenWidth	String	O	Total width of the cardholder's screen in pixels. Example: 1536.
javaEnabled	String	O	A Boolean value that represents the ability of the cardholder browser to execute Java. Value is returned from the navigator.javaEnabled property. Possible Values:True/False
browserIpAddress	String	O	Browser IP Address
screenHeight	String	O	Total height of the Cardholder's scree in pixels, example: 864.
browserName	String	O	Value of the User-Agent header sent by the client's web browser.
timeDifference	String	O	Time difference between UTC time and the cardholder browser local time, in minutes, Example:300
colorDepth	String	O	Value represents the bit depth of the color palette for displaying images, in bits per pixel.
contentType	String	O	content type of the HTTP accept header.
jsEnabled	String	O	A Boolean value that represents the ability of the cardholder browser to execute JavaScript. Possible Values:True/False.

Parameter Name	Data Type	(M/O/C)	Description
browserEmail	String	O	Note: Merchants should be able to know the values from fingerprint details of cardholder's browser. Send the same value as email. Alphanumeric string without special characters, except allowed email characters, 1-100 in length

3.4.2 Header Parameter

Parameter Name	Data Type	(M/O/C)	Description
Authorization	String	M	SHA 256 hash value of the JSON request body; encrypted with a predefined key.

Note: The predefined key is shared to the merchant during merchant onboarding.

3.4.3 Response Parameters

Response Parameter	Description
status	Transaction status
authorizedAmount	Amount authorized
totalAmount	Total amount
currency	3 letter ISO currency code
transactionId	TerraPay PG internal reference number
merchantId	TerraPay PG merchant ID
remittanceId	Incoming remittance ID
partnerImtTxnRefno	Incoming transaction ref Id
partnerImtTxnAuthCode	Incoming transaction auth Id
tokenCard	Token number sent by merchant

Sample success response

```
{
   "transactionId": "100000020765",
   "status": "AUTHORIZED",
   "authorizedAmount": "100",
   "totalAmount": "100",
   "currency": "ISK",
   "txnRefNum": "227809869994",
   "remittanceId": "QR037IZJJEMNZ3Q20",
   "merchantId": "650715952301812",
   "tokenCard": {
      "tokenPAN": "ea789781-91a1-4b5f-937a-cb875d19c12d"
   },
   "partnerImtTxnAuthCode": "QR037IZJJEMNZ3Q20",
   "partnerImtTxnRefno": "ea1ff362-6e52-45f3-b235-9bf65f2730c3"
}
```

3.5 Error Codes

Error Code	Description	Comments
TP100	Invalid merchant code	Should exist in the TerraPay PG system
TP101	Invalid remittance Id	Should be a valid UUID, if present
TP102	Invalid address	
TP103	Invalid city	
TP104	Invalid country	Should be a valid ISO country code
TP105	Invalid email	
TP106	Invalid first name	
TP107	Invalid last name	
TP108	Invalid phone number	
TP109	Invalid postal code	
TP110	Invalid currency	Should be a valid ISO currency code
TP111	Invalid amount	Should be positive without currency separators
TP500	Generic system error	Any TerraPay PG server error will be trapped and responded.
TP510	Network system error	Any Network error will be processed and responded.
TP520	Network timeout error	
M5-A	Wrong Currency Code.	
05-I	Do not honor.	
14-I	Invalid card number.	
AF-A	Invalid CardVD.	

Error Code	Description	Comments
400	400 Bad Request	
TXN_R8	BIN not configured : This card cannot be used for this transaction	
TP105	Card Authentication error	
TimeOut	Authentication timed out	
TP_119	RemittanceId unexpected for non-remittance transactions.	

3.6 Status Check API

Status check API can be used by merchant to get the current status of a transaction. Merchant can pass either transactionId or remittanceId or partnerImtTxnRefNo with merchantId to statusCheck API to fetch the details of a transaction.

Merchant has to use the API key shared with them after onboarding, to encrypt the card details. The key issued will be an AES 256 key. Algorithm used to encrypt the card details should be AES/ECB/PKCS5Padding.

Transaction	Status Check
URL	https://<IP_Address>:8045/txn-service/eComm/statusCheck
Method	POST
Content Type	application/json

3.6.1 Request Parameters

Parameter Name	Data Type	(M/O/C)	Description
merchantId	String	M	Alphanumeric string without special characters, 1-10 in length.
remittanceId	String	C	Alphanumeric string received from remittance platform. 16 characters in length. This value will be available only for remittance transactions.
partnerImtTxnRefno	String	C	This is the merchant transaction reference id
transactionId	String	C	Transaction Id sent by payment gateway in the response to merchant

3.6.2 Request Header

Parameter Name	Data Type	(M/O/C)	Description
authorization	String	M	SHA 256 hash value of the JSON request body; encrypted with a predefined key. Encryption algo to be used: AES/ECB/PKCS5Padding

Sample Request

```
{
   "merchantId”: 614415553564066,
   "remittanceId”: "TP345655464646",
   "transactionId": "100000696288",
   "partnerImtTxnRefno": "354364574576"
}
```

Sample Response

```
{
   "transactionId": "100000696288",
   "status": "AUTHORIZED",
   "authorizedAmount": "56.23",
   "currency": "ISK",
   "terraPayIMTTxnRef": "TP345653453454646",
   "authProcessedTime": "16-01-2023 21:31:00",
   "authReceivedTime": "16-01-2023 21:31:00",
   "partnerImtTxnRefno": "354364574576"
}
```

3.6.3 Transaction Status in TerraPay PG

1. In-Progress: Transaction has IN-PROGRESS status when PG has received the request and PG is yet to initiate Authentication process.
2. In-Verification: If transaction status is “In-Verification”, then it denotes that the authentication response has not been received by payment gateway or authentication is in progress.
3. In-Authorization: If transaction status is “In-Authorization”, then it denotes that the authentication is successful and payment gateway has initiated authorization and it is in progress.
4. Authorized: If transaction status is “Authorized”, then it denotes that the authorization request was successful.
5. Declined: If transaction status is “Declined”, then it denotes that the authorization request was not successful.