



VegaaH Payment Gateway Application

API Specification Document

Version 3.0

Concerto Software & Systems Pvt Ltd.
06-May-2024

DOCUMENT VERSION

This section lists the details of reviews for this document:

Version No.	Name of Reviewer	Role	Date
3.0	Vishal Utekar	Technical Lead	06-May-2024

CHANGE HISTORY

This section details the changes this document has undergone:

Version No.	Details of Change	Changed Sections	Date
3.0	Base Version		06-May-2024

Table of Contents

1 Overview	3
2 Payment Features Available	4
3 Logical Components	5
4. Hosted Payment Page Integration	6
4.1 Payment Method Supported -	6
3.1 Transactions supported -	7
3.2 Headers and Authentication	7
3.2.1 Pre-Authorization and Purchase Transaction Flow	7
3.2.2 Recurring Transactions	10
3.2.3 Tokenization Transactions	13
3.2.4 STC Pay Transactions	21
3.3 UPI Purchase Transactions	23
3.4 NetBanking Purchase Transactions	25
3.5 Wallet Purchase Transactions	27
4 API Based Transaction Direct Transactions	29
4.1 Transactions Supported	29
4.2 Purchase/Pre-authorization Non-3D Transaction	30
4.3 Purchase/ Pre-authorization 3D Transaction	33
4.4 Recurring Transactions	36
4.5 Relevant Transactions. VOID Purchase/ VOID Pre-authorization/ Capture/ VOID Capture/ Refund/ VOID Refund	41
4.6 Inquiry Transactions	44
4.7 Standalone refund Transactions	47
4.8 Excess refund Transactions	50
4.9 Excess Capture Transactions	51
4.10 STCPay Direct Transactions	59
4.11 Apple Pay Direct Transactions	61
4.12 Payment Link Transactions	65
4.13 Add Tokenization Transactions	69
5 How to create request Hash or Signature use in the request and response	74
6 How to encrypt and decryption merchant final response	75
7 Transactions with Action Codes	77
8. Instrument type description	77
9. Country Codes and description	77
10. Currency Codes and description	78
11. Response Mapping on receipt Page	78
12. APPROVALS	78

1 Overview

VegaaH is e-commerce payment processing gateway for the merchants. The purpose of VegaaH payment gateway is to make the merchant's online e-commerce business up and running with the help of simple integration.

This document is intended for use by developers from Merchant side who need guidance on interfacing with VegaaH Payment Gateway Application. It describes the interface that merchant system uses to process the e-commerce transactions using standard form posting method. This document describes various integration methods as Transport method, Hosted payment page method, direct payment method and 3D secure processing method.

2 Payment Features Available

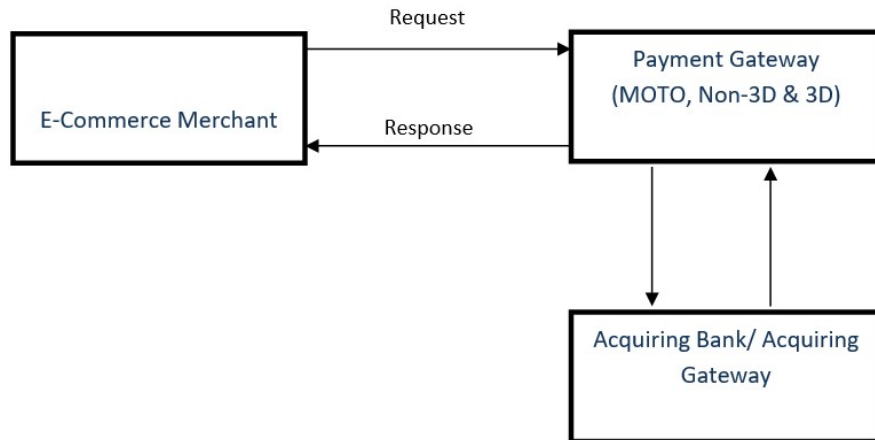
Payment Methods	Transaction Types	GUI	API
Credit Card/Debit Card	Purchase	Y	Y
	Pre-authorization	Y	Y
	Capture	Y	Y
	Refund	Y	Y
	Void	Y	Y
	Inquiry	N	Y
	Recurring	Y	Y
	Token	N	Y
	Token with purchase	N	Y
	Standalone Refund	N	Y
	Excess Refund	Y	Y
STC Pay	Purchase	N	Y
	Refund	Y	Y
	Inquiry	N	Y
	Void Purchase	Y	Y
	Void Refund	Y	Y
SADAD	Purchase	Y	Y
	Inquiry	N	Y
	Cancel	Y	Y
APPLE PAY	Purchase	N	Y
	Pre-authorization	N	Y
	Capture	Y	Y
	Refund	Y	Y
	Void	Y	Y
	Inquiry	N	Y
Payment Link	Purchase	Y	Y
	Pre-authorization	Y	Y
Invoice Payment	Purchase	Y	N
	Pre-authorization	Y	N

Note: The above transactions are initiated through the GUI and are supported when initiated from the Merchant Portal.

3 Logical Components

VegaaH payment gateway exchanges data between merchant and gateway in the form of Request and Response.

The following diagram explains the components involved in the process.



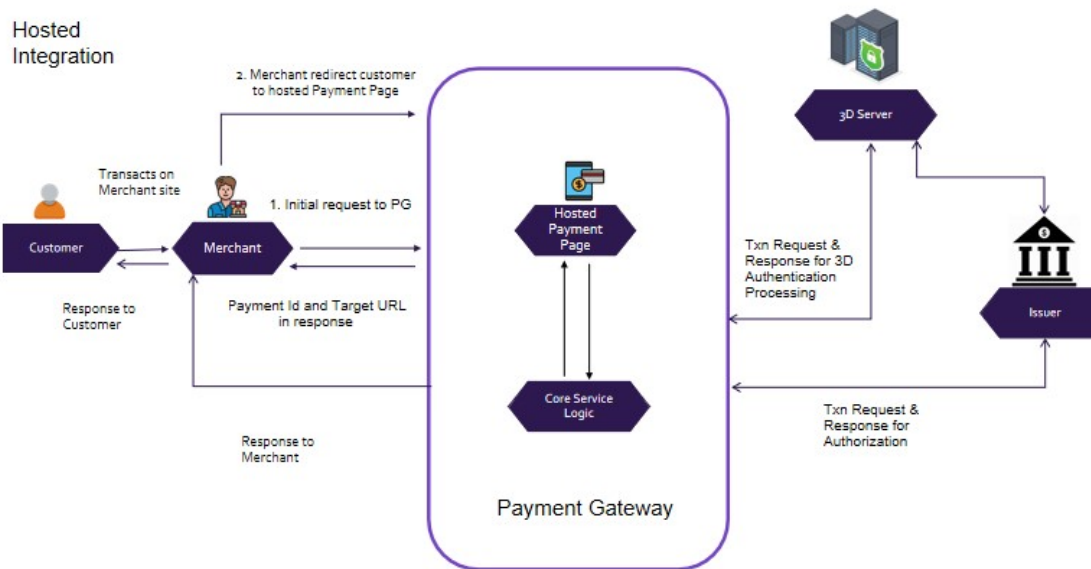
Merchant integrated with Gateway always initiates the transaction by sending request to gateway by HTTPS POST method. Gateway replies with the response over the same socket to merchant. Response time depends on the acquiring bank/ acquiring gateway processing with the inter-bank networks.

We recommend to set the timeout request connection for 60 seconds.

Request/Response process is identified by the set of JSON message fields which provides the transaction interface between the merchant and the VegaaH payment gateway. Each integration URL is given below.

4. Hosted Payment Page Integration

Hosted payment page method is an operation to extract card details in a secured manner as it is on this page that the customer enters sensitive card details. This page is hosted by payment gateway or the acquiring gateway. In this method merchant cannot store card details on their system.



4.1 Payment Method Supported -

- ✓ Credit Card
- ✓ Debit Card
- ✓ Tokenization (Card On File)
- ✓ STC Pay
- ✓ UPI
- ✓ Net Banking
- ✓ Wallet
- ✓ SADAD
- ✓ APPLEPAY

3.1 Transactions supported -

- ✓ Purchase Transaction -
- ✓ Pre-authorization Transaction
- ✓ Recurring Transaction
- ✓ Tokenization (Card On File)

There are two legs of transaction in this integration:

Leg 1 - Initiate the request to PG to fetch the payment id and target URL

Leg 2- Merchant should do the form post to target URL with payment id received in response. Customer enters the card details on the hosted page and submits the request. Gateway will process the request further for authorization and once gateway will receive the response from bank, it will send it to merchant receipt URL

3.2 Headers and Authentication

The headers and authentication for various transaction types are as below

3.2.1 Pre-Authorization and Purchase Transaction Flow

The merchant initiates a transaction request with details such as Terminal ID, Track ID, Amount, Payment Type, Currency, and customer information like email ID, address, and other mandatory fields specified in the API request below.

After submitting the request, the merchant will receive a response containing the transaction ID. Using the payment link URL and transaction ID, the merchant should redirect the customer to the payment page, where the customer can securely enter their card details and submit the payment. The gateway will process the authentication and authorization, and once the bank responds, the gateway will forward the outcome to the merchant's receipt URL, if configured.

Post Request by Merchant on PG URL:

PGRequestURL:{domain-name}}/{pg.context.path}}/v2/payments/pay-request

Merchant has to do form post on below URL (Merchant redirection) :

Redirect URL(payment link URL) : {domain-name}}/{pg.context.path}}/direct.htm?paymentId=

API request field details:

<u>Name</u>	<u>Length</u>	<u>Mandatory/Optional/Conditional</u>	<u>Description</u>
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction

order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
order.description		Optional	Additional information about the order
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description, e.g Currency Names available in the section currency code and description
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Optional	Customer email address
customer.mobileNumber	12	Conditional	Customer contact number
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide additional information in JSON format, which will be stored on the gateway

Sample request for payment ID generation:

Initiate the request to fetch the payment ID and customer redirect URL -

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "e344f6b8d80ddc02c6a96e6d8e4c274f1070330d31ccc5ebe77b322526b07abc",
  "paymentType": "4",    // "1" for purchase transaction
  "amount": "10.00",
```

```

"currency": "SAR",
"order": {
  "orderId": "TCMPGSHT3D0001",
  "description": "Purchase of product XYZ"
},
"customer": {
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"additionalDetails": {
  "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

API response field details:

Name	Type	Description
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
transactionId	String	Unique Transaction generated by payment gateway
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
paymentLink.linkUrl	String	URL of hosted payment page where merchant redirect customer
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request
amountDetails.originalAmount	String	Original transaction amount send in request

Sample response for payment ID generation -

```
{
  "responseCode": "001",
  "responseDescription": "Transaction Approved",
  "transactionId": "2411316219097943174",
  "transactionDateTime": "April 22, 2024 4:14 PM",
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "paymentLink": {
    "linkUrl": "{{domain-name}}/{{pg.context.path}}/direct.htm?paymentId="
  },
  "terminalId": "TestTerm",
  "amountDetails": {
    "amount": "10.00",
    "originalAmount": "10.00"
  }
}
```

3.2.2 Recurring Transactions**API request field details:**

<u>Name</u>	<u>Length</u>	<u>Mandatory/ Optional/Co nditional</u>	<u>Description</u>
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
order.description		Optional	Additional information about
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway.
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secrete key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2

currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description, e.g Currency Names available in the section currency code and description
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Optional	Customer email address
Customer.mobileNumber	12	Optional	Customer contact number
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide addition information in JSON format, which will stored on the gateway
subscriptionSchedule.frequency		Mandatory	Subscription frequency. Valid values will be D - Daily W - Weekly M - Monthly Y - Yearly Q - QUARTERLY HALFY - Half Yearly
subscriptionSchedule.startDate		Mandatory	Start date of the subscription in <dd/MM/yyyy> format.
subscriptionSchedule.subscriptionAmount		Mandatory	Subscription amount of the transaction with or without decimal points i.e 2.25 or 2
subscriptionSchedule.subscriptionType		Mandatory	Type of subscription "s" for subscription and "I" for installation and "NS" for NO Subscription.
subscriptionSchedule.paymentDays		Conditional	<p>If subscriptionSchedule.frequency is set to "W" (weekly), the subscriptionSchedule.paymentDays can be any day of the week (e.g., Sunday, Monday).</p> <p>For subscriptionSchedule.frequency set to "M" (monthly), "Y" (yearly), "Q" (quarterly), or "HALFY" (half-yearly), the subscriptionSchedule.paymentDays must fall between the 1st and 28th of the month.</p> <p>If subscriptionSchedule.frequency is set to "D" (daily), there is no specified payment day.</p>

subscriptionSchedule. paymentType		Mandatory	"R" for recurring.
subscriptionSchedule. noOfSubscriptionPay ments		Mandatory	The number of times the payment subscription will continue.

Purchase Recurring Request Sample:

```

{
  "terminalId": "TestTerminal",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "1", // "4" for pre-authorization
  "amount": "100.00",
  "currency": "SAR",
  "order": {
    "orderId": "123456",
    "description": "Purchase of product XYZ"
  },
  "customer": {
    "customerEmail": "test.customer@concertosoft.com",
    "billingAddressStreet": "Ghatkopar",
    "billingAddressCity": "MUMBAI",
    "billingAddressState": "MAHARASHTRA",
    "billingAddressPostalCode": "400075",
    "billingAddressCountry": "IN"
  },
  "subscriptionSchedule": {
    "frequency": "W",
    "startDate": "30/05/2024",
    "subscriptionAmount": "50",
    "subscriptionType": "i",
    "paymentDays": "WEDNESDAY",
    "paymentType": "R",
    "noOfSubscriptionPayments": "2"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
transactionId	String	Unique Transaction generated by payment gateway
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
paymentLink.linkUrl	String	URL of hosted payment page where merchant redirect customer
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request
amountDetails.originalAmount	String	Original transaction amount send in request

Sample response for payment ID generation -

```
{
  "responseCode": "001",
  "responseDescription": "Transaction Approved",
  "transactionId": "2411316219097947654",
  "transactionDateTime": "April 12, 2024 4:19 PM",
  "orderDetails": {
    "orderId": "TCMPGSHT3D9812"
  },
  "paymentLink": {
    "linkUrl": "{{domain-name}}/{{pg.context.path}}/direct.htm?paymentId="
  },
  "terminalId": "TestTerm",
  "amountDetails": {
    "amount": "5.00",
    "originalAmount": "5.00"
  }
}
```

3.2.3 Tokenization Transactions

VegaaH PG Card-On-File integration allows merchants to tokenize customer card data within the payment gateway. This enables customers to initiate payment transactions using tokenized cards

instead of re-entering their card details during each payment. Customers can register their card details through the merchant's website in the payment gateway for re-use during future payments.

Merchants need to store the token, the masked card number, and the registered customer's unique ID. This way, when customers make subsequent purchases or conduct financial transactions on the merchant's website, they can select the previously tokenized card to complete the process.

Tokenization Purchase Request Sample -

API request field details:

Name	Length	Mandatory/ Optional/Conditional	Description
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction 12 - for token creation
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
order.description		Optional	Additional information about
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway.
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Optional	Customer email address

ail				
Customer.mobileNumber	12	Optional		Customer contact number
additionalDetails.userData	String	200	Optional	User defined field. User/Merchant can provide addition information in JSON format, which will stored on the gateway
tokenization.cardToken	String	50	Conditional	This is mandatory for “Token with purchase” transaction and if values of “tokenization.operation” is “U” or “D”
tokenization.operation	String	1	Conditional	This is mandatory if value of “paymentType” field is set to “12” for below token operation. A : To add token U: To update the token D: To delete token

Sample request for Add Tokenization Transaction -

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "12",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "M11",
    "description": "Purchase of product XYZ"
  },
  "tokenization": {
    "operation": "A"
  },
  "customer": {
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "additionalDetails": {
    "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
  }
}
```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

Sample response for Add Tokenization Transaction -

```
{
  "responseCode": "001",
  "responseDescription": "Transaction Approved",
  "transactionId": "2411316219097947654",
  "transactionDateTime": "April 12, 2024 4:19 PM",
  "orderDetails": {
    "orderId": "TCMPGSH3D9845"
  },
  "paymentLink": {
    "linkUrl": "{{domain-name}}/{{pg.context.path}}/direct.htm?paymentId="
  },
  "terminalId": "TestTerm",
  "amountDetails": {
    "amount": "5.00",
    "originalAmount": "5.00"
  }
}
```

Sample request for Update Tokenization Transaction -

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "12",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "M11",
    "description": "Purchase of product XYZ"
  },
  "tokenization": {
    "cardToken": <clear token from merchant> // mandatory if paymentType=1/4 OR tokenization.operation =D/U
    "operation": "U"
  },
  "customer": {
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",

```

```

    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "additionalDetails": {
    "userData": "{\"entryone\":\"abc\\\", \"entrytwo\":\"def\\\", \"entrythree\":\"xyz\\\"}"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

Sample response for Update Tokenization Transaction -

```

{
  "responseCode": "001",
  "responseDescription": "Transaction Approved",
  "transactionId": "2411316219097947654",
  "transactionDateTime": "April 12, 2024 4:19 PM",
  "orderDetails": {
    "orderId": "TCMPGSHT3D9845"
  },
  "paymentLink": {
    "linkUrl": "{{domain-name}}/{{pg.context.path}}/direct.htm?paymentId="
  },
  "terminalId": "TestTerm",
  "amountDetails": {
    "amount": "5.00",
    "originalAmount": "5.00"
  },
  "tokenization": {
    "cardToken": "4111112503501111"
  }
}

```

Sample request for Delete Tokenization Transaction -

```

{

```

```

"terminalId": "TestTerm",
"password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
"paymentType": "12",
"amount": "10.00",
"currency": "SAR",
"order": {
  "orderId": "M11",
  "description": "Purchase of product XYZ"
},
"tokenization": {
  "cardToken": "<clear token from merchant> // mandatory if paymentType=1/4 OR tokenization.operation =D/U",
  "operation": "D"
},
"customer": {
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"additionalDetails": {
  "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

Sample response for delete token -

```

{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": "Saved card deleted successfully",
  "transactionId": "2413712423054002904",
  "transactionDateTime": "May 16, 2024 12:48 PM",
  "orderDetails": {
    "orderId": "M11"
  },
  "terminalId": "TestTerm",

```

```

"signature": "33fccba2ef0ccac044179954bfdabe7c1f247cfac88d7f6bbcca5f9d5f4c8dbc",
"amountDetails": {
  "amount": "1.00",
  "originalAmount": "1.00"
},
"tokenization": {
  "cardToken": "4111112503501111"
}
}

```

Tokenization Purchase Request Sample -

```

{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "1",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "M11",
    "description": "Purchase of product XYZ"
  },
  "card": {
    "cvv": "123"
  },
  "tokenization": {
    "cardToken": "4111117997401111"
  },
  "customer": {
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "additionalDetails": {
    "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

Sample response for Tokenization Purchase Transaction -

```
{
  "responseCode": "001",
  "responseDescription": "Transaction Approved",
  "transactionId": "2411316219097947654",
  "transactionDateTime": "April 12, 2024 4:19 PM",
  "orderDetails": {
    "orderId": "TCMPGSHT3D9845"
  },
  "paymentLink": {
    "linkUrl": "{{domain-name}}/{{pg.context.path}}/direct.htm?paymentId="
  },
  "terminalId": "TestTerm",
  "amountDetails": {
    "amount": "5.00",
    "originalAmount": "5.00"
  },
  "tokenization": {
    "cardToken": "4111112503501111"
  }
}
```

API response field details:

Name	Type	Description
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
transactionId	String	Unique Transaction generated by payment gateway
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
paymentLink.linkUrl	String	URL of hosted payment page where merchant redirect customer
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request

amountDetails.originalAmount	String	Original transaction amount send in request
tokenization.cardToken	String	Card token on which update/delete operation performed

3.2.4 STC Pay Transactions

Note: This transaction is not supported by the PG_INDIA setup.

Initiate the request to fetch the payment id and target URL - The request is same as mentioned in [5.3.1](#)

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
signature	String	This is a signature generated based on HMAC and merchant secrete key. The merchant can validate it to check the authenticity of the the response.
merchantLogo	String	Merchant logo in base64 encoded form.
merchantReferenceId	String	Merchant Id
transactionId	String	Unique transaction ID generated by the payment gateway
transactionDateTime	String	Transaction date and time logged at the payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amou nt	String	Transaction amount including surcharge if any.
amountDetails.origin alAmount	String	Original transaction amount send in request
currency	String	Transaction currency code

tokenization.cardToken	String	Tokenized card which merchant can save for future use.
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.
subscriptionSchedule.subscriptionId	String	Subscription Id generated after successfully creation of subscription in payment gateway
customerName	String	Customer Name received in transaction request
customerEmail	String	Customer Email Id address
rrn	String	Retrieval Reference Number
merchantName	String	Merchant name
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT

Sample decrypted final merchant response-

```
{
  "transactionDateTime": "May 16, 2024 12:47 PM",
  "signature": "a121e0c94e1b2bb9b06dfc9fd1b2dfa6695dd4fd805ec9d213eba73728fa6a2b",
  "merchantLogo": "",
  "merchantReferenceId": "MERDEF003844",
  "transactionId": "2413712422958685901",
  "amountDetails": {
    "amount": "1.00",
    "originalAmount": "1.00"
  },
  "customerName": "",
  "responseCode": "000",
  "rrn": "",
  "merchantName": "Test Merchant",
  "result": "SUCCESS",
  "orderDetails": {
    "orderId": "123456"
  },
  "responseDescription": "Transaction Successful",
  // For token start
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "tokenization": {
```

```

    "cardToken": "4111112503501111"
  },
  // For token end
  // subscription start
  "subscriptionSchedule": {
    "subscriptionId": "2413713423763436836"
  }
  //subscription end
  "customerEmail": "merchant.autouser@concentosoft.com",
  "currency": "SAR"
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

3.3 UPI Purchase Transactions

Initiate the request to fetch the payment id and target URL - The request is same as mentioned in [5.3.1](#)

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the

API response field details:

Name	Type	Description
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
merchantLogo	String	Merchant logo in base64 encoded form.
merchantReferenceId	String	Merchant Id
transactionId	String	Unique transaction ID generated by the payment gateway
transactionDateTime	String	Transaction date and time logged at the payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount including surcharge if any.

amountDetails.originalAmount	String	Original transaction amount send in request
currency	String	Transaction currency code
customerEmail	String	Customer Email Id address
rrn	String	Retrieval Reference Number
merchantName	String	Merchant name
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT

Sample decrypted final merchant response-

```
{
  "transactionDateTime": "May 16, 2024 12:47 PM",
  "signature": "a121e0c94e1b2bb9b06dfc9fd1b2dfa6695dd4fd805ec9d213eba73728fa6a2b",
  "merchantLogo": "",
  "merchantReferenceId": "MERDEF003844",
  "transactionId": "2413712422958685901",
  "amountDetails": {
    "amount": "1.00",
    "originalAmount": "1.00"
  },
  "customerName": "",
  "responseCode": "000",
  "rrn": "",
  "merchantName": "Test Merchant",
  "result": "SUCCESS",
  "orderDetails": {
    "orderId": "123456"
  },
  "responseDescription": "Transaction Successful",
  // For token start
  "cardDetails": {
    "maskedCard": "411111XXXXX1111",
    "cardBrand": "VISA"
  },
  "tokenization": {
    "cardToken": "4111112503501111"
  },
  // For token end
  // subscription start
  "subscriptionSchedule": {
```

```

        "subscriptionId": "2413713423763436836"
    }
    //subscription end
    "customerEmail": "merchant.autouser@concertosoft.com",
    "currency": "SAR"
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

3.4 NetBanking Purchase Transactions

Initiate the request to fetch the payment id and target URL - The request is same as mentioned in [5.3.1](#)

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the

API response field details:

Name	Type	Description
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
signature	String	This is a signature generated based on HMAC and merchant secrete key. The merchant can validate it to check the authenticity of the the response.
merchantLogo	String	Merchant logo in base64 encoded form.
merchantReferenceId	String	Merchant Id
transactionId	String	Unique transaction ID generated by the payment gateway
transactionDateTime	String	Transaction date and time logged at the payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amou nt	String	Transaction amount including surcharge if any. E.g 101.00
amountDetails.origin alAmount	String	Original transaction amount send in request e. g100.00
currency	String	Transaction currency code
customerEmail	String	Customer Email Id address

rrn	String	Retrieval Reference Number
merchantName	String	Merchant name
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT

Sample decrypted final merchant response-

```

{
  "transactionDateTime": "May 16, 2024 12:47 PM",
  "signature": "a121e0c94e1b2bb9b06dfc9fd1b2dfa6695dd4fd805ec9d213eba73728fa6a2b",
  "merchantLogo": "",
  "merchantReferenceId": "MERDEF003844",
  "transactionId": "2413712422958685901",
  "amountDetails": {
    "amount": "1.00",
    "originalAmount": "1.00"
  },
  "customerName": "",
  "responseCode": "000",
  "rrn": "",
  "merchantName": "Test Merchant",
  "result": "SUCCESS",
  "orderDetails": {
    "orderId": "123456"
  },
  "responseDescription": "Transaction Successful",
  // For token start
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "tokenization": {
    "cardToken": "4111112503501111"
  },
  // For token end
  // subscription start
  "subscriptionSchedule": {
    "subscriptionId": "2413713423763436836"
  },
  //subscription end
  "customerEmail": "merchant.autouser@concertosoft.com",
  "currency": "SAR"
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

3.5 Wallet Purchase Transactions

Initiate the request to fetch the payment id and target URL - The request is same as mentioned in [5.3.1](#)

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
merchantLogo	String	Merchant logo in base64 encoded form.
merchantReferenceId	String	Merchant Id
transactionId	String	Unique transaction ID generated by the payment gateway
transactionDateTime	String	Transaction date and time logged at the payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount including surcharge if any. E.g 101.00
amountDetails.originalAmount	String	Original transaction amount send in request e.g 100.00
currency	String	Transaction currency code
customerEmail	String	Customer Email Id address
rrn	String	Retrieval Reference Number
merchantName	String	Merchant name

result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
--------	--------	---

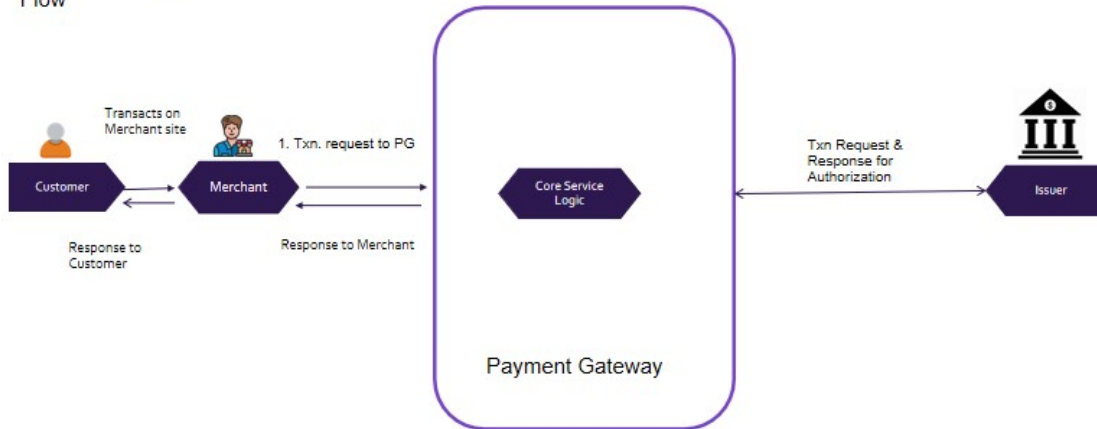
Sample decrypted final merchant response-

```
{
  "transactionDateTime": "May 16, 2024 12:47 PM",
  "signature": "a121e0c94e1b2bb9b06dfc9fd1b2dfa6695dd4fd805ec9d213eba73728fa6a2b",
  "merchantLogo": "",
  "merchantReferenceId": "MERDEF003844",
  "transactionId": "2413712422958685901",
  "amountDetails": {
    "amount": "1.00",
    "originalAmount": "1.00"
  },
  "customerName": "",
  "responseCode": "000",
  "rrn": "",
  "merchantName": "Test Merchant",
  "result": "SUCCESS",
  "orderDetails": {
    "orderId": "123456"
  },
  "responseDescription": "Transaction Successful",
  // For token start
  "cardDetails": {
    "maskedCard": "411111XXXXX1111",
    "cardBrand": "VISA"
  },
  "tokenization": {
    "cardToken": "4111112503501111"
  },
  // For token end
  // subscription start
  "subscriptionSchedule": {
    "subscriptionId": "2413713423763436836"
  },
  //subscription end
  "customerEmail": "merchant.autouser@concertosoft.com",
  "currency": "SAR"
}
```

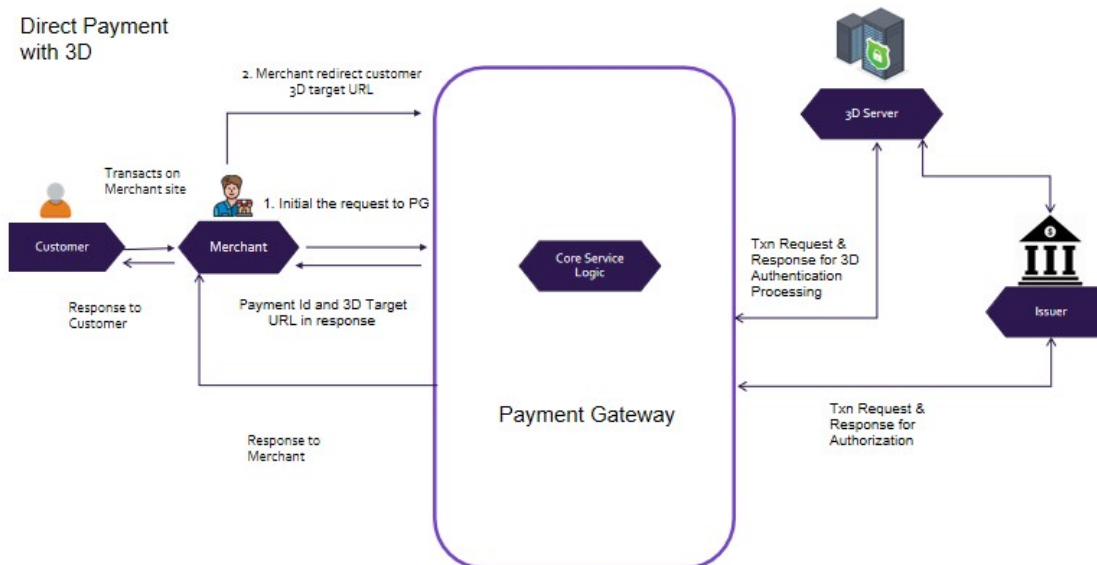
NOTE : To create the signature hash or response hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

4 API Based Transaction Direct Transactions

Direct Transaction Flow



Direct Payment with 3D



4.1 Transactions Supported

- ✓ Purchase Transaction
- ✓ Purchase 3D Transaction
- ✓ Pre-authorization Transaction
- ✓ Refund Transaction

- ✓ Capture Transaction
- ✓ Void Transaction
- ✓ Standalone Refund Transaction
- ✓ Excess Refund Transaction
- ✓ Recurring Transaction (Purchase and Pre-authorization)
- ✓ Payment Link Creation
- ✓ SADAD Payment Creation
- ✓ SADAD Payment Cancel
- ✓ Tokenization (Card On File)
- ✓ STCPAY Payment

There is one leg of transaction in this integration. Each transaction request send by merchant will get the response from gateway on same socket or connection.

4.2 Purchase/Pre-authorization Non-3D Transaction

API request field details:

Name	Length	Mandatory/ Optional/ Conditional	Description
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
card.number	19	Mandatory	Plain card no entered by customer on merchant portal
card.expiryMonth		Mandatory	Card expiry month in MM format
card.expiryYear		Mandatory	Card expiry year in YYYY format
card.cvv	3-4	Mandatory	Card CVV number
paymentInstrument.paymentMethod		Mandatory	Valid values will be CCI and DCI, etc. As provided in section Instrument type provided.
customer.cardHolderN	100	Mandatory	Card holder name

ame			
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Conditional	Customer email address
customer.mobileNumber	12	Conditional	Customer contact number
userData	200	Optional	User defined field. User/Merchant can provide addition information in JSON format, which will stored on the gateway

Purchase Transaction Request Sample -

```

{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "e344f6b8d80ddc02c6a96e6d8e4c274f1070330d31ccc5ebe77b322526b07abc",
  "paymentType": "1", // 4 for pre-authorization
  "amount": "10.00",
  "currency": "SAR",
  "card": {
    "number": "<plain card number>",
    "expiryMonth": "07",
    "expiryYear": "2029",
    "cvv": "123"
  },
  "order": {
    "orderId": "TCMPGSHT3D0001"
  },
  "customer": {
    "cardHolderName": "FirstName LastName",
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
}
```



```

    "paymentInstrument": {
      "paymentMethod": "DCI"
    }
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it.

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
transactionId	String	Unique transaction ID generated by the payment gateway
authCode	String	Auth code received from the transaction processor
transactionDateTime	String	Transaction date and time logged at the payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount including surcharge if any. Amount of the transaction with or without decimal points i.e 2.25 or 2
amountDetails.originalAmount	String	Original transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
currency	String	Transaction currency code
tokenization.cardToken	String	Tokenized card which merchant can save for future use.
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.
customerName	String	Customer Name received in transaction request
customerEmail	String	Customer Email Id address
rrn	String	Retrieval Reference Number

result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
--------	--------	---

Purchase Sample Response:

```
{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2411318293410495322",
  "transactionDateTime": "April 22, 2024 6:17 PM",
  "authCode": "326363",
  "rrn": "411312326363",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "terminalId": "TestTerm",
  "signature": "ce9ca8623add12b3f67fdc34dfab9947ab112e185c4272929fd94dca19cbc3fa",
  "amountDetails": {
    "amount": "10.00",
    "originalAmount": "1.00"
  }
}
```

4.3 Purchase/ Pre-authorization 3D Transaction

This is a two legs of transaction in this integration:

Leg 1 - Initiate the request to PG to fetch the payment id and target URL

Leg 2- Merchant should do the form post to target URL with payment id received in response to process 3D transaction, Customer will redirect to 3D secure page if steps are successfully passed other wise it will redirect customer to receipt page.

API request field details:

<u>Name</u>	<u>Length</u>	<u>Mandatory/Optional/Conditional</u>	<u>Description</u>
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction

order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
card.number	19	Mandatory	Plain card no entered by customer on merchant portal
card.expiryMonth		Mandatory	Card expiry month in MM format
card.expiryYear		Mandatory	Card expiry year in YYYY format
card.cvv	3-4	Mandatory	Card CVV number
paymentInstrument.paymentMethod		Mandatory	Valid values will be CCI and DCI, etc. As provided in section Instrument type provided.
customer.cardHolderName	100	Mandatory	Card holder name
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Conditional	Customer email address
customer.mobileNumber	12	Conditional	Customer contact number
userData	200	Optional	User defined field. User/Merchant can provide additional information in JSON format, which will be stored on the gateway

Purchase/Pre-authorization Transaction Request Sample -

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
```

```

"signature": "e344f6b8d80ddc02c6a96e6d8e4c274f1070330d31ccc5ebe77b322526b07abc",
"paymentType": "1", // 4 - for pre-authorization
"amount": "10.00",
"currency": "SAR",
"card": {
  "number": "<plain card number>",
  "expiryMonth": "07",
  "expiryYear": "2029",
  "cvv": "123"
},
"order": {
  "orderId": "TCMPGSHT3D0001"
},
"customer": {
  "cardHolderName": "FirstName LastName",
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"paymentInstrument": {
  "paymentMethod": "DCI"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
transactionId	String	Unique Transaction generated by payment gateway
transactionDateTime	String	Transaction date and time logged at payment gateway

orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
paymentLink.linkUrl	String	URL of hosted payment page where merchant redirect customer
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
amountDetails.originalAmount	String	Original transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2

Purchase/Pre-authorization 3D Sample Response:

```
{
  "responseCode": "001",
  "responseDescription": "Transaction Approved",
  "transactionId": "2411414102035394988",
  "transactionDateTime": "April 23,2024 2:29 PM",
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "paymentLink": {
    "linkUrl": "{{domain-name}}/{{pg.context.path}}/3dredirect.htm?paymentId="
  },
  "terminalId": "TestTerm",
  "amountDetails": {
    "amount": "10.00",
    "originalAmount": "10.00"
  }
}
```

4.4 Recurring Transactions

API request field details:

<u>Name</u>	<u>Length</u>	<u>Mandatory/Optional/Conditional</u>	<u>Description</u>
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters

order.description		Optional	Recurring subscription description for which it is getting created.
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
card.number	19	Mandatory	Plain card no entered by customer on merchant portal
card.expiryMonth		Mandatory	Card expiry month in MM format
card.expiryYear		Mandatory	Card expiry year in YYYY format
card.cvv	3-4	Mandatory	Card CVV number
paymentInstrument.paymentMethod		Mandatory	Valid values will be CCI and DCI, etc. As provided in section Instrument type provided.
customer.cardHolderName	100	Mandatory	Card holder name
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Conditional	Customer email address
customer.mobileNumber	12	Conditional	Customer contact number
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide additional information in JSON format, which will be stored on the gateway
subscriptionSchedule.frequency		Mandatory	D - Daily W - Weekly M - Monthly Y - Yearly Q - QUARTERLY

			HALFY - Half Yearly
subscriptionSchedule.startDate		Mandatory	Start date of the subscription in <dd/MM/yyyy> format.
subscriptionSchedule.subscriptionAmount		Mandatory	Subscription amount of the transaction with or without decimal points i.e 2.25 or 2
subscriptionSchedule.subscriptionType		Mandatory	Type of subscription "S" for subscription and "I" for installation and "NS" for NO Subscription.
subscriptionSchedule.paymentDays		Conditional	<p>If subscriptionSchedule.frequency is set to "W" (weekly), the subscriptionSchedule.paymentDays can be any day of the week (e.g., Sunday, Monday).</p> <p>For subscriptionSchedule.frequency set to "M" (monthly), "Y" (yearly), "Q" (quarterly), or "HALFY" (half-yearly), the subscriptionSchedule.paymentDays must fall between the 1st and 28th of the month.</p> <p>If subscriptionSchedule.frequency is set to "D" (daily), there is no specified payment day.</p>
subscriptionSchedule.paymentType		Mandatory	"R" for recurring.
subscriptionSchedule.noOfSubscriptionPayments		Mandatory	The no. of times payment subscription will be continued.

Recurring Purchase Sample Request -

```

{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "1",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "M11",
    "description": "Purchase of product XYZ"
  },
  "customer": {
    "cardHolderName": "Test test",
    "customerEmail": "merchantuser@concertosoft.com",
    "billingAddressCity": "MUMBAI",
    "billingAddressPostalCode": "400075",

```

```

    "billingAddressCountry": "IN"
  },
  "paymentInstrument": {
    "paymentMethod": "DCI"
  },
  "card": {
    "number": "<plain card number>",
    "expiryMonth": "12",
    "expiryYear": "2025",
    "cvv": "123"
  },
  "subscriptionSchedule": {
    "frequency": "W",
    "startDate": "23/04/2024",
    "subscriptionAmount": "5",
    "subscriptionType": "s/i/ns",
    "paymentDays": "Wednesday",
    "paymentType": "R",
    "noOfSubscriptionPayments": "2"
  },
  "additionalDetails": {
    "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

API response field details:

Name	Type	Description
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
signature	String	This is a signature generated based on HMAC and merchant secrete key. The merchant can validate it to check the authenticity of the the response.
transactionId	String	Unique Transaction generated by payment gateway

rrn	String	Retrieval Reference Number
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
amountDetails.originalAmount	String	Original transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
subscriptionSchedule.subscriptionId	String	Subscription Id generated after successfully creation of subscription in payment gateway
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.

Recurring Purchase Sample Response-

```
{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": "Transaction Successful",
  "transactionId": "2413713424398148475",
  "transactionDateTime": "May 16, 2024 1:11 PM",
  "authCode": "113030",
  "rrn": "413707113030",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "M11"
  },
  "subscriptionSchedule": {
    "subscriptionId": "2413713424401118364"
  },
  "terminalId": "TestTerm",
  "signature": "48564288c8717bebadf737557d6c8f9eadd0a84b849593a88b9ddf403d2e47dc",
  "amountDetails": {
    "amount": "10.00",
    "originalAmount": "10.00"
  }
}
```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

4.5 Relevant Transactions. VOID Purchase/ VOID Pre-authorization/ Capture/ VOID Capture/ Refund/ VOID Refund

API request field details:

<u>Name</u>	<u>Length</u>	<u>Mandatory/Optional/Conditional</u>	<u>Description</u>
paymentType	2	Mandatory	2 - Refund 5 - Capture 3 - VOID purchase 6 - VOID refund 7 - VOID Capture 9 - VOID Pre-authorization
referenceId	50	Mandatory	Original transaction Id to be pass as reference Id in relevant transaction
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secrete key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
paymentInstrument.paymentMethod		Optional	Valid values will be CCI and DCI, etc. As provided in section Instrument type provided.
customer.cardHolderName	100	Mandatory	Card holder name

customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.mobileNumber	50	Conditional	Customer email address
mobileNumber		Conditional	Customer contact number
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide addition information in JSON format, which will stored on the gateway

Void Request Sample -

```

{
  "referenceId": "2413712423366196453",
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "0fa4a5718d7e7a1e74f4140bb004a33f4e557e939d19d2dbe87ce57004dc9d5c",
  "paymentType": "3", // 9 -VOID Pre-authorization
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "TCMPGSHT3D0001"
  },
  "customer": {
    "cardHolderName": "FirstName LastName",
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "paymentInstrument": {
    "paymentMethod": "DCI"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

API request field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
signature	String	This is a signature generated based on HMAC and merchant secrete key. The merchant can validate it to check the authenticity of the the response.
transactionId	String	Unique Transaction generated by payment gateway
rrn	String	Retrieval Reference Number
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
amountDetails.originalAmount	String	Original transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
subscriptionSchedule.subscriptionId	String	Subscription Id generated after successfully creation of subscription in payment gateway
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.

Void Transactions Response Sample –

```

{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2413713424775807911",
  "transactionDateTime": "May 16, 2024 1:17 PM",
  "authCode": "114110",
  "rrn": "413707110930",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "terminalId": "TestTerm",
  "signature": "e2ac8398ca90091ca66c42c3e09dc54553808e8eb00b07a5ed37fcbe61993b2f",
  "amountDetails": {
    "amount": "10.00",
    "originalAmount": "10.00"
  }
}

```

4.6 Inquiry Transactions**API request field details:**

<u>Name</u>	<u>Length</u>	<u>Mandatory/ Optional/Conditional</u>	<u>Description</u>
paymentType	2	Mandatory	10 - Transaction inquiry
referenceId	50	Mandatory	Original transaction Id to be pass as reference Id in relevant transaction

order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secrete key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
paymentInstrument.paymentMethod		Optional	Valid values will be CCI and DCI, etc. As provided in section Instrument type provided.
customer.cardHolderName	100	Optional	Card holder name
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Conditional	Customer email address
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide addition information in JSON format, which will stored on the gateway

Inquiry Request Sample -

```
{
  "referenceId": "2411016209313643227",
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "0fa4a5718d7e7a1e74f4140bb004a33f4e557e939d19d2dbe87ce57004dc9d5c",
  "paymentType": "10",
  "amount": "10.00",
  "currency": "SAR",
```

```

"order": {
  "orderId": "TCMPGSHT3D0001"
},
"customer": {
  "cardHolderName": "Abhijit Bhattacharya",
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"paymentInstrument": {
  "paymentMethod": "DCI"
},
"additionalDetails": {
  "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Inquiry Response Sample -

API response field details:

Name	Type	Description
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
transactionId	String	Unique Transaction generated by payment gateway
rrn	String	Retrieval Reference Number
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2

amountDetails.originalAmount	String	Original transaction amount send in request.Amount of the transaction with or without decimal points i.e 2.25 or 2
subscriptionSchedule.subscriptionId	String	Subscription Id generated after successfully creation of subscription in payment gateway
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.

```
{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": "SUCCESS",
  "transactionId": "2411516194400967144",
  "transactionDateTime": "2024-04-24 16:06:20.057",
  "authCode": "",
  "rrn": "",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "terminalId": "TestTerm",
  "signature": "98fc6262525ccc26436461fb207ff737dc481335e181fe132c012b48d9250f6a",
  "amountDetails": {
    "amount": "10.0000",
    "originalAmount": "10.00"
  }
}
```

4.7 Standalone refund Transactions

Standalone Refund Sample Request - (Using Token)

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "314a2b266eb6dc391864b50620079f4f9f42982c367891f97c7e06e42058dfb2",
  "paymentType": "14",
  "amount": "10.00",
  "currency": "SAR",
```



```

"tokenization": {
  "cardToken": "4141424973903071508"
},
"order": {
  "orderId": "TCMPGSHT3D0023"
},
"customer": {
  "cardHolderName": "cardholder name",
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"paymentInstrument": {
  "paymentMethod": "CCI"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Standalone Refund Sample Response –

```

{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2411516194542797704",
  "transactionDateTime": "April 24, 2024 4:11 PM",
  "authCode": "",
  "rrn": "",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "TCMPGSHT3D0023"
  },
  "terminalId": "TestTerm",
  "signature": "64e8df6a48103b152a2ad549576dbfd5a069f6fdb6c2a975f4306f065a123c6b",
  "amountDetails": {
    "amount": "10.00"
  },
  "tokenization": {
    "cardToken": "4141424973903071508"
  }
}

```

```
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Standalone Refund Sample Request - (Using card)

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "314a2b266eb6dc391864b50620079f4f9f42982c367891f97c7e06e42058dfb2",
  "paymentType": "14",
  "amount": "10.00",
  "currency": "SAR",
  "card": {
    "number": "5123450000000008",
    "expiryMonth": "07",
    "expiryYear": "2029",
    "cvv": "123"
  },
  "order": {
    "orderId": "TCMPGSHT3D0023"
  },
  "customer": {
    "cardHolderName": "cardholder name",
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "paymentInstrument": {
    "paymentMethod": "CCI"
  }
}
```

Standalone Refund Sample Response –

```
{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2411516194676270544",
  "transactionDateTime": "April 24, 2024 4:13 PM",
  "authCode": "",
  "rrn": ""
}
```

```

"cardDetails": {
  "maskedCard": "512345XXXXXX0008",
  "cardBrand": "MasterCard"
},
"orderDetails": {
  "orderId": "TCMPGSHT3D0023"
},
"terminalId": "TestTerm",
"signature": "ee64cac9bc4148914fb04a7d9f5a9bd8a8651cd37b1613c7b4b71bdf2182f402",
"amountDetails": {
  "amount": "10.00"
}
}

```

4.8 Excess refund Transactions

Excess Refund Sample Request -

Note - Excess Refund transaction is only work with acquirer.

```

{
  "referenceId": "2411510175859360343",
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "0fa4a5718d7e7a1e74f4140bb004a33f4e557e939d19d2dbe87ce57004dc9d5c",
  "paymentType": "2",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "TCMPGSHT3D0001"
  },
  "customer": {
    "cardHolderName": "FirstName LastName",
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "paymentInstrument": {
    "paymentMethod": "DCI"
  },
  "internalProcessing": {
    "identifierFlag": "ER"
  },
  "additionalDetails": {

```

```

    "userData": "{\\"entryone\\":\\"abc\\",\\"entrytwo\\":\\"def\\",\\"entrythree\\":\\"xyz\\"}"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Excess Refund Sample Response –

```

{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2411511176162341302",
  "transactionDateTime": "April 24, 2024 11:04 AM",
  "authCode": "",
  "rrn": "",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "paymentLink": {
    "linkBaseFlag": "N"
  },
  "terminalId": "TestTerm",
  "signature": "2fcf82fa04eb9852a8e3ba8b46c3dd19070a9264e5e727284e5206134c13e29f",
  "amountDetails": {
    "amount": "10.00"
  }
}

```

4.9 Excess Capture Transactions

Excess Refund Sample Request -

Note - Excess Capture transaction is only work with acquirer.

```

{
  "referenceId": "2411510175859360343",
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "0fa4a5718d7e7a1e74f4140bb004a33f4e557e939d19d2dbe87ce57004dc9d5c",
  "paymentType": "5",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "TCMPGSHT3D0001"
  }
}

```

```

},
"customer": {
  "cardHolderName": "FirstName LastName",
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"paymentInstrument": {
  "paymentMethod": "DCI"
},
"additionalDetails": {
  "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Excess Capture Sample Response –

```

{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2411511176162341302",
  "transactionDateTime": "April 24, 2024 11:04 AM",
  "authCode": "",
  "rrn": "",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "paymentLink": {
    "linkBaseFlag": "N"
  },
  "terminalId": "TestTerm",
  "signature": "2fcf82fa04eb9852a8e3ba8b46c3dd19070a9264e5e727284e5206134c13e29f",
  "amountDetails": {
    "amount": "10.00"
  }
}

```

SADAD Transactions

Note: This transaction is not supported by the PG_INDIA setup.

API request field details:

<u>Name</u>	<u>Length</u>	<u>Mandatory/ Optional/Conditional</u>	<u>Description</u>
paymentType	2	Mandatory	1 - Purchase Transaction
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
paymentInstrument.paymentMethod		Mandatory	Value will be "SADAD"
paymentInstrument.channelName		Mandatory	Value will be "SADAD"
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Conditional	Customer email address
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide additional information in JSON format, which will be stored on the gateway

sadadPaymentDetails. customerFullName	60	Mandatory	Customer Full Name (i.e First Name and Last Name)
sadadPaymentDetails. districtCode		Optional	Customer District Code
sadadPaymentDetails. customerEmailAddresses	50	Mandatory	Customer Email ID
sadadPaymentDetails. cityCode		Optional	Customer City Code
sadadPaymentDetails. postalCode	10	Optional	Customer Postal Code
sadadPaymentDetails. miniPartialAmount		Conditional	If Partial Payment is set to "Y", then the Customer Minimum Partial Amount must be greater than 10.
sadadPaymentDetails. customerPreviousBalance	10	Conditional	Customer Previous Amount
sadadPaymentDetails. serviceName	20	Optional	Service Name
sadadPaymentDetails. customerIdNumber	20	Optional	Customer Id Number
sadadPaymentDetails. streetName	20	Optional	Customer Street Name
sadadPaymentDetails. billItemList		Mandatory	Customer Bill List
sadadPaymentDetails. entityActivityId		Optional	Activity Id
sadadPaymentDetails. buildingNumber	20	Optional	Customer Building Name
sadadPaymentDetails. isPartialAllowed		Mandatory	Partial Flag (i.e Y/N)
sadadPaymentDetails. customerMobileNumber	12	Mandatory	Customer Mobile Number

sadadPaymentDetails. expireDate		Mandatory	Sadad Payment Expiry Date Format YYYY-MM-DD	
sadadPaymentDetails. customerIdType	20	Optional	Customer Id Type	
sadadPaymentDetails. sendLinkMode		Mandatory	"SendLinkMode" specifies the preferred mode of sending, whether it's via email or SMS. 0 - SMS 1 - EMAIL 2 - BOTH	
sadadPaymentDetails. smsLanguage	String		optional	"SMSLanguage" determines the language preference for sending messages, whether it's English ("en") or Arabic ("ar").
sadadPaymentDetails. smsTemplate	String	250	optional	"SMSTemplate" requires a customized SMS template to be sent.
merchantIp	String		optional	Merchant IP address
customerIp	String		optional	Customer IP address

SADAD Transaction Sample Request-

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "07c9b25793442689f398dd30368a8b4a0498ea458fdb862b9b24eb936d0adb41",
  "paymentType": "1",
  "amount": "150.00",
  "currency": "SAR",
  "paymentInstrument": {
    "paymentMethod": "SADAD",
    "channelName": "SADAD"
  },
},
"customer": {
  "cardHolderName": "FirstName LastName",
  "customerEmail": "customer.Email@concertosoft.com",
  "billingAddressStreet": "R.B. Street",
  "billingAddressCity": "MUMBAI",
  "billingAddressState": "MAHARASHTRA",
  "billingAddressPostalCode": "400075",
  "billingAddressCountry": "IN"
},
"sadadPaymentDetails": {
  "customerFullName": "FirstName LastName",
```



```

    "districtCode": "1",
    "customerEmailAddress": "customer.email@concertosoft.com",
    "cityCode": "12",
    "postalCode": "22230",
    "miniPartialAmount": "10.00",
    "customerPreviousBalance": "",
    "serviceName": "Fintech Service",
    "customerIdNumber": "2023222934",
    "streetName": "R.B STREET",
    "billItemList":
    [{"unitPrice": "150.00", "quantity": "1", "name": "shoes", "vat": "0", "discount": "0", "discountType": "
    FIXED"}],
    "entityActivityId": "547",
    "buildingNumber": "1234",
    "isPartialAllowed": "Y",
    "customerMobileNumber": "966557778988",
    "expireDate": "2023-11-11",
    "customerIdType": "NAT",
    "billNumber": "",
    "sendLinkMode": "2",
    "smsLanguage": "en",
    "smsTemplate": ""
  },
  "order": {
    "orderId": "M11"
  },
  "additionalDetails": {
    "userData": ""
  },
  "merchantIp": "192.168.1.1",
  "customerIp": "192.168.1.2",
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

API response field details:

Name	Type	Description
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
result	String	INITIATED
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
transactionId	String	Unique Transaction generated by payment gateway

rrn	String	Retrieval Reference Number
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request
amountDetails.originalAmount	String	Original transaction amount send in request
subscriptionSchedule.subscriptionId	String	Subscription Id generated after successfully creation of subscription in payment gateway
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.
sadadPaymentDetails.billNumber	String	SADAD bill number
sadadPaymentDetails.sadadNumber	String	SADAD number

SADAD transaction sample response:

```
{
  "responseCode": "5005",
  "result": "INITIATED",
  "responseDescription": ":Got Sadad Number Successfully",
  "transactionId": "2411511176334587398",
  "transactionDateTime": "April 24, 2024 11:07AM",
  "authCode": "",
  "rrn": "",
  "orderDetails": {
    "orderId": "M11"
  },
  "sadadPaymentDetails": {
    "billNumber": "24012021024",
    "sadadNumber": "809407"
  },
  "terminalId": "TestTerm",
  "signature": "6be06093d3034f145656aee2cf3b55b4205958e722dd5ad5131411bfac02a63",
  "amountDetails": {
    "amount": "150.00"
  }
}
```

```

}

```

SADAD Payment Cancel Sample Request-

Note: This transaction is not supported by the PG_INDIA setup.

API request field details:

URL to be used to initiate cancel SADAD: {url-context}/sadam/cancelSadamTransaction

<u>Name</u>	<u>Length</u>	<u>Mandatory/ Optional/Conditional</u>	<u>Description</u>
billNumber	50	Mandatory	SADAD bill number received in initial request
sadamNumber	30	Mandatory	SADAD number received in initial request

```

{
  "billNumber": "24012021024", // Bill number received in Initial Request
  "sadamNumber": "845763" // Sadad number received in Initial Request
}

```

SADAD Payment Cancel successful Response –

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
Message	String	Response code of the transaction
Result	String	Response code description

```

{
  "Message": "Operation Done Successfully",
  "Result": "success"
}

```

SADAD Payment Cancel Failure Response –

```

{
  "Message": "Cancellation not allowed for this transaction.",
  "Result": "failure"
}

```

4.10 STCPay Direct Transactions

Note: This transaction is not supported by the PG_INDIA setup.

The merchant will receive a response containing the transaction ID. The merchant should use the payment link URL, transaction ID, and redirect the customer to the payment link URL. Here, the customer can enter their OTP on the secure payment page and submit the request. The gateway will then process the request for authentication and authorization. Once the gateway receives the response from the bank, it will send the response to the merchant's receipt URL, if configured.

Merchant has to do form post on below URL (Merchant redirection) :

RedirectURL(paymentlinkURL){domain-name}}/{pg.context.path}}/STC3DRedirect.jsp?paymentID
=

API request field details:

Name	Length	Mandatory/ Optional/Conditional	Description
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
order.description		Optional	Additional information about the order
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description, e.g Currency Names available in the section currency code and description
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled

customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Optional	Customer email address
customer.mobileNumber	12	Mandatory	Customer contact number
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide additional information in JSON format, which will be stored on the gateway

```

{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "e344f6b8d80ddc02c6a96e6d8e4c274f1070330d31ccc5ebe77b322526b07abc",
  "paymentType": "1",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "TCMPGSHT3D0001",
    "description": "Purchase of product XYZ"
  },
  "customer": {
    "mobileNumber": "557877988",
    "customerEmail": "merchant.autouser@concertosoft.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "paymentInstrument": {
    "paymentMethod": "STCPAY",
    "channelName": "STCPAY"
  },
  "additionalDetails": {
    "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
transactionId	String	Unique Transaction generated by payment gateway
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
paymentLink.linkUrl	String	URL of hosted payment page where merchant redirect customer
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request
amountDetails.originalAmount	String	Original transaction amount send in request

Sample response -

```
{
  "responseCode": "001",
  "responseDescription": "Transaction Approved",
  "transactionId": "2411316219097943174",
  "transactionDateTime": "April 22, 2024 4:14 PM",
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "paymentLink": {
    "linkUrl": "{{domain-name}}/{{pg.context.path}}/STC3DRedirect.jsp?paymentId="
  },
  "terminalId": "TestTerm",
  "amountDetails": {
    "amount": "10.00",
    "originalAmount": "10.00"
  }
}
```

4.11 Apple Pay Direct Transactions

Note: This transaction is not supported by the PG_INDIA setup.

Note: This API request will not be triggered directly via the API, as it requires an Apple Pay token. The request provided is only a sample initiated through plugins for demonstration purposes.

API request field details:

<u>Name</u>	<u>Length</u>	<u>Mandatory/ Optional/Conditional</u>	<u>Description</u>
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
order.description		Optional	Additional information about the order
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description, e.g Currency Names available in the section currency code and description
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Optional	Customer email address
customer.mobileNumber	12	Mandatory	Customer contact number

additionalDetails.user Data	200	Optional	User defined field. User/Merchant can provide addition information in JSON format, which will stored on the gateway
additionalDetails.deviceInfo		Optional	Only in Plugin Through transcation initiated
PaymentToken		Mandatory	In only Direct Applepay Transction

```

{
  "terminalId": "TestTerm",
  "password": "Password@123",
  "signature": "0fa4a5718d7e7a1e74f4140bb004a33f4e557e939d19d2dbe87ce57004dc9d5c",
  "paymentType": "1",
  "amount": "1.00",
  "currency": "SAR",
  "card": {
    "number": "xxxxxxxxxxxxxxxx",
    "expiryMonth": "07",
    "expiryYear": "2029",
    "cvv": "xxx"
  },
  "order": {
    "orderId": "TCMPGSHT3D0001"
  },
  "customer": {
    "cardHolderName": "test test",
    "customerEmail": "merchant.autouser@gmail.com",
    "billingAddressStreet": "101 Mahape",
    "billingAddressCity": "Mumbai",
    "billingAddressState": "Maharashtra",
    "billingAddressPostalCode": "400709",
    "billingAddressCountry": "SA"
  },
  "paymentInstrument": {
    "paymentMethod": "APPLEPAY",
    "channelName": "APPLEPAY"
  },
  "merchantIP": "10.10.11.37",
  "PaymentToken": "<Applepay Token>",
  "additionalDetails": {
    "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}",
    "deviceInfo": "{\"pluginVersion\":\"3.0.0\",\"pluginName\":\"JAVA_HOSTED\"}"
  }
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Final merchant response:

This response is in encrypted form and merchant need to [decrypt](#) it. This response will send to the merchant on merchant's receipt URL.

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
transactionId	String	Unique Transaction generated by payment gateway
rrn	String	Retrieval Reference Number
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
amountDetails.originalAmount	String	Original transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
subscriptionSchedule.subscriptionId	String	Subscription Id generated after successfully creation of subscription in payment gateway
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.

APPLEPAY Direct Transactions Response Sample –

```
{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2413713424775807911",
  "transactionDateTime": "May 16, 2024 1:17 PM",
  "authCode": "114110",
  "rrn": "413707110930",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "TCMPGSHT3D0001"
  },
  "terminalId": "TestTerm",
  "signature": "e2ac8398ca90091ca66c42c3e09dc54553808e8eb00b07a5ed37fcbe61993b2f",
  "amountDetails": {
    "amount": "10.00",
    "originalAmount": "10.00"
  }
}
```

4.12 Payment Link Transactions**Payment Link API -****API request field details:**

Name	Length	Mandatory/ Optional/Co nditional	Description
paymentType	15	Mandatory	15 - for Payment Link Creation

order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
order.description		Optional	Additional information about the order
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description, e.g Currency Names available in the section currency code and description
customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Optional	Customer email address
customer.cardHolderName	100	Mandatory	Card holder name
customer.mobileNumber	12	Mandatory	Customer contact number
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide additional information in JSON format, which will be stored on the gateway
additionalDetails.deviceInfo		Optional	Only in Plugin Through transaction initiated
paymentLink.purpose		Mandatory	Purpose of link Creation

paymentLink.expiryDate		Mandatory	Link Expiry Date
paymentLink.allowPartialPayment		Mandatory	Partial Value will be Y/N
paymentLink.linkType		Mandatory	Link Type i.e. API

Payment Link Creation Sample Request-

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "15",
  "amount": "10.00",
  "currency": "SAR",
  "order": {
    "orderId": "M11",
    "description": "Purchase of product XYZ"
  },
  "customer": {
    "cardHolderName": "FirstName LastName",
    "customerEmail": "customer.email@concertosoft.com",
    "billingAddressCity": "MUMBAI",
    "billingAddressPostalCode": "400075",
    "billingAddressCountry": "IN"
  },
  "paymentLink": {
    "purpose": "shopping",
    "expiryDate": "20240202",
    "allowPartialPayment": "N",
    "linkType": "a"
  }
}
```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
transactionId	String	Unique Transaction generated by payment gateway
rrn	String	Retrieval Reference Number
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
amountDetails.originalAmount	String	Original transaction amount send in request. Amount of the transaction with or without decimal points i.e 2.25 or 2
subscriptionSchedule.subscriptionId	String	Subscription Id generated after successfully creation of subscription in payment gateway
cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.

Payment Link Creation Sample Response –

```
{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": "Transaction Successful",
  "transactionId": "2411516195760305256",
  "transactionDateTime": "April 24, 2024 4:17 PM",
  "orderDetails": {
    "orderId": "M11"
  },
  "terminalId": "TestTerm",
  "signature": "41839ebffdeec073c0d6647898a38b2069fab8ce7b73e86d56b54b8122d982f",
}
```

```

    "amountDetails": {
      "amount": "10.00"
    }
  }
}

```

4.13 Add Tokenization Transactions

Add Tokenization Transaction Request Sample -

API request field details:

Name	Length	Mandatory/ Optional/Conditional	Description
paymentType	2	Mandatory	1 - for purchase transaction 4 - for pre-authorization transaction 12 - for tokenized request
order.orderId	50	Mandatory	Unique merchant order Id provided by the merchant application to identify the transaction on gateway. Avoid using space and special characters
order.description		Optional	Additional information about
terminalId	50	Mandatory	Field indicates the unique terminal id provided by Gateway administrator. This field is required for payment gateway to identify the merchant on gateway.
password	100	Mandatory	Field indicates the password associated with the terminal id provided to the merchant. It is required to authenticate the merchant on gateway
signature		Mandatory	This is a signature generated based on HMAC and merchant secret key
amount		Mandatory	Amount of the transaction with or without decimal points i.e 2.25 or 2
currency		Mandatory	Alphabetic currency code of the transaction, e.g Currency Names available in the section currency code and description
card.number	19	Mandatory	Plain card no entered by customer on merchant portal
card.expiryMonth		Mandatory	Card expiry month in MM format
card.expiryYear		Mandatory	Card expiry year in YYYY format
card.cvv	3-4	Mandatory	Card CVV number
paymentInstrument.paymentMethod		Mandatory	Valid values will be CCI and DCI, etc. As provided in section Instrument type provided.

customer.billingAddressPostalCode	15	Conditional	Customer Postal code. This field is mandatory if AVS check is enabled
customer.billingAddressCountry		Conditional	Customer Country in ISO country code Alpha-2. This field is mandatory if AVS check is enabled
customer.billingAddressCity	30	Conditional	Customer city. This field is mandatory if AVS check is enabled
customer.billingAddressState	30	Conditional	Customer state. This field is mandatory if AVS check is enabled
customer.billingAddressStreet	150	Conditional	Customer address. This field is mandatory if AVS check is enabled
customer.customerEmail	50	Optional	Customer email address
Customer.mobileNumber	12	Optional	Customer contact number
additionalDetails.userData	200	Optional	User defined field. User/Merchant can provide addition information in JSON format, which will stored on the gateway
tokenization.cardToken	50	Conditional	This is mandatory for "Token with purchase" transaction and if values of "tokenization.operation" is "U" or "D"
tokenization.operation	1	Conditional	This is mandatory if value of "paymentType" field is set to "12" for below token operation. A : To add token U: To update the token D: To delete token

Sample request for add token:

```

{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "12",
  "amount": "10.00",
  "currency": "SAR",
  "card": {
    "number": "<plain card number>",
    "expiryMonth": "07",
    "expiryYear": "2029",
    "cvv": "123"
  },
  "order": {
    "orderId": "M11"
  },
  "tokenization": {
    "operation": "A"
  },
}

```

```

"customer": {
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"paymentInstrument": {
  "paymentMethod": "DCI"
},
"additionalDetails": {
  "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Add Tokenization Transaction Sample Response –

API response field details:

<u>Name</u>	<u>Type</u>	<u>Description</u>
responseCode	String	Response code of the transaction
responseDescription	String	Response code description
result	String	Transaction result, its value will be SUCCESS, FAILURE or TIMEOUT
signature	String	This is a signature generated based on HMAC and merchant secret key. The merchant can validate it to check the authenticity of the response.
transactionId	String	Unique Transaction generated by payment gateway
rrn	String	Retrieval Reference Number
transactionDateTime	String	Transaction date and time logged at payment gateway
orderDetails.orderId	String	Unique merchant order Id provided by the merchant application in payment request.
terminalId	String	Same terminal id sent in request
amountDetails.amount	String	Transaction amount send in request
amountDetails.originalAmount	String	Original transaction amount send in request

cardDetails.maskedCard	String	Customer's masked card number
cardDetails.cardBrand	String	Card brand like VISA, MasterCard, Amex etc.
tokenization.cardToken	String	Tokenized card which merchant can save for future use.

Sample response for add token:

```
{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": "Saved card <updated/deleted> successfully",
  "transactionId": "2413713425575132351",
  "transactionDateTime": "May 16, 2024 1:30 PM",
  "cardDetails": {
    "maskedCard": "411111XXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "M11"
  },
  "terminalId": "TestTerm",
  "signature": "ef51222c229287d07695064c35a840ccd023ca40a2ec4a50f57fa3419babe240",
  "amountDetails": {
    "amount": "1.00",
    "originalAmount": "1.00"
  },
  "tokenization": {
    "cardToken": "4111118748281111"
  }
}
```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Tokenization Purchase Request Sample -

```
{
  "terminalId": "TestTerm",
  "password": "TerminalPassword",
  "signature": "caa93ce1e3896342d156fdb9aacdc79b3ca2422e3c2ac6c82c60c9245ba70d48",
  "paymentType": "1",
  "amount": "10.00",
  "currency": "SAR",
```

```

"order": {
  "orderId": "M11",
  "description": "Purchase of product XYZ"
},
"card": {
  "cvv": "123"
},
"tokenization": {
  "cardToken": "4111117997401111"
},
"customer": {
  "customerEmail": "merchant.autouser@concertosoft.com",
  "billingAddressStreet": "101 Mahape",
  "billingAddressCity": "Mumbai",
  "billingAddressState": "Maharashtra",
  "billingAddressPostalCode": "400709",
  "billingAddressCountry": "SA"
},
"paymentInstrument": {
  "paymentMethod": "DCI"
},
"additionalDetails": {
  "userData": "{\"entryone\":\"abc\",\"entrytwo\":\"def\",\"entrythree\":\"xyz\"}"
}
}

```

NOTE : To create the signature hash or response hash used in the above request, refer to the [Create signature hash or response hash Method](#).

Tokenization Purchase Response Sample -

```

{
  "responseCode": "000",
  "result": "SUCCESS",
  "responseDescription": ":Transaction Successful",
  "transactionId": "2411516195611945907",
  "transactionDateTime": "April 24, 2024 4:29 PM",
  "authCode": "002765",
  "rrn": "411510002765",
  "cardDetails": {
    "maskedCard": "411111XXXXXX1111",
    "cardBrand": "VISA"
  },
  "orderDetails": {
    "orderId": "M11"
  },
  "paymentLink": {
    "linkBaseFlag": "N"
  }
}

```

```

    },
    "terminalId": "TestTerm",
    "signature": "cbdf5289e88c53296e3b2b8601eb25aecad7b633f67651d2978d5c0283ee454b",
    "amountDetails": {
      "amount": "10.00"
    },
    },
    "tokenization": {
      "cardToken": "4111117997401111"
    }
  }
}

```

5 How to create request Hash or Signature use in the request and response

Signature use in the request and response field is created will be created with the below format using SHA256 algorithm to create the hash.

Format: trackId|terminalId|password|mechantkey|amount|currency

Sample:

M11_Track|TestTest|password|85a74e068216737f6c0e829fb6b87839301bf20219b8df053ac7cc1e611133d3|100.00|SAR

String pipeSeperatedString =

M11_Track+"|"+TestTest+"|"+password+"|"+85a74e068216737f6c0e829fb6b87839301bf20219b8df053ac7cc1e611133d3+"|"+1.00+"|"+SAR;

String reqSignature= hash.sha256(pipeSeperatedString);

How to create response signature/hash:

Signature use in the response will be created with the below format using SHA256 algorithm to create the hash.

Format: PaymentId|merchantkey|responseCode|amount

Sample:

1234567891234567891|85a74e068216737f6c0e829fb6b87839301bf20219b8df053ac7cc1e611133d3|**000**|**100.00**

```
String pipeSeperatedString =request.getAttribute("TranId")+"|" +merchantKey+"|" +
request.getAttribute("ResponseCode")+"|" +tran.getTdDtranAmount().setScale(2,
RoundingMode.CEILING).toString();
```

```
String rspSignature= hash.sha256(pipeSeperatedString);
```

Test Credentials:

- 1) Purchase voucher Credit card number :

Card Holder Name: Test Test (should be same as first leg request)

Card Number: 4111 1111 1111 1111

Card Expiry: 10/2028 (date should be greater than current date)

CVV: 123

6 How to encrypt and decryption merchant final response

Encryption and Decryption Process

Merchant final response posted on merchant receipt URL is encrypted using below snippet.

To encrypt the merchant's final response, the following logic is implemented:

```
public String getEncryptedResponse(String jsonResponse, String merKey) {
    String encryptedResponse = "";
    try {
        // Convert merchant key to SecretKey
        SecretKey secretKey = new SecretKeySpec(hexStringToByteArray(merKey), "AES");

        // Encrypt and encode the JSON data
        encryptedResponse = encryptAndEncode(jsonResponse, secretKey);
    } catch (Exception e) {
        // Handle exception
    }
    return encryptedResponse;
}
```

Explanation:

- 1 Convert the hexadecimal string representation of the merchant key (**merKey**) into a **SecretKey** object.

- 2 Use the **SecretKey** to encrypt the JSON response.
- 3 Return the encrypted response.

Decryption Logic

To decrypt the encrypted response, the following code snippet can be use:

Below is Encrypted response in String :

String

```
encryptedResponse= "data=g5wxHqQvmDx62Bcc%2FTz3B2jbckD3IKiy%2F9qhWohPf37RZ9NYccC
NYrap%2B3eA14bp1RgeuZNNHJqYliqrcQz7u8XfTe%2FEArGV2%2BIBdfDfdq6fVFjufF2mqI8Y6uUOK
6FgH1q6phtjr1v4Ulag2JgNxqLOaFNd4wOKf%2BqfSihw5ssc4f3OCi6WFB2lZ52llie8VAKUMFt1%2B
h12mCufUmqEMHiJHDEcgqHRTJgv1b8d2lQLKY8KUfDJD4xm%2Bkptck%2FrRNIGwzjojq3qt3LbY3
GEF7B4aeNLuJslogsAfpFS37uYwnKMFqA8GCq%2BSWRcG%2Fdt%2FxrIjsz1Vdc94tNa6JbMIQ0DB
6wYyQZc7jRb7ZWxJANRTB7lI1sRYZd3N7wY8XVDB%2Fgtq0%2FZVBii2Zqt83yLdt5xTvqOnk68fLYKF
Lz6sPwiPNKdRM60KHBXJmN9nhifh1sjWksMyiCtQu1HPKGtEnegpok4eaamP1XVJWgC40J%2Fr7oC
z5zhNHKK2asDpCipCo19%2FqU%2BCD0ECRx%2B1KQWcbu8%2BdgnMQnKB2hjAxavWuYj8lhTIVA
x%2BcvCS4a%2BKOGCTowMSmARKiuRpO%2FEQmwKcR98s3EDRq7C86ZWvNExp3Wswfxz1vISZx
DbogJgjtCg5RTcFN2ZaCsRNczIM0uWyHkYYU%2FYag%2BKmD%2FhC6bChxQJhXS9OMuafxy5udfA
Ny9Hp2FVGt%2BtpvCoMVKho%2B4wxIEiANKBlqaPFJQYYF6ya2YFgkQLfm1Rdh344L2004nUPx6t7
JknWrNmrQnQ23xHsjw3fWe44h7H6Yz3TwQuKSZWXXeVmv87YkOrdO8cqj5J5AhIN%2Bvg1VHdzlZ
dpUctQ%3D%3D&termId=Testterm";
```

```
public String decodeAndDecryptV2(String encryptedResponse, String merKey) throws
Exception {
    // Convert merchant key to SecretKey
    SecretKey secretKey = new SecretKeySpec(hexStringToByteArray(merKey), "AES");

    // Decode and decrypt the encrypted response
    String decodedAndDecrypted = decodeAndDecrypt(encryptedResponse, secretKey);

    return decodedAndDecrypted;
}
```

Explanation:

- 1 Convert the hexadecimal string representation of the merchant key (**merKey**) into a **SecretKey** object.
- 2 Use the **SecretKey** to decrypt and decode the encrypted response.
- 3 Optionally, print the decrypted data.
- 4 Return the decrypted and decoded response.

7 Transactions with Action Codes

Transaction	Action Code
Purchase	1
Refund	2
Void Purchase	3
Pre-authorization	4
Capture	5
Void Refund	6
Void Capture	7
Void Pre-authorization	9
Transaction Inquiry	10
Tokenization	12
Standalone Refund	14
Payment Link	15

8. Instrument type description

Instrument Type	Description
CCI	Credit Card
DCI	Debit Card
SADAD	SADAD
STCPAY	STCPAY
APPLE PAY	APPLE PAY
UPI	UPI
NETBANK	NETBANKING
WALLET	WALLET

9. Country Codes and description

Country Name	Country Code
Saudi Arabia	SA
Bahrain	BH
Jordan	JO

Kuwait	KW
Oman	OM
Qatar	QA
Yemen	YE
India	IN
United States	US
United Kingdom	GB
United Arab Emirates	UAE
Iraq	IQ

10. Currency Codes and description

Currency Name	Currency Code	Description
INR	356	Indian Rupee
AFN	514	Afghan Afghani
EUR	978	Euro
IQD	365	Iraqi Dinar
SAR	682	Saudi Riyal
USD	840	US Dollar

11. Response Mapping on receipt Page

Vegaah will send the response data to merchant receipt URL (Merchant receipt URL will configure in Vegaah).

Merchant has to get the response from request parameter.

JAVA/JSP :

12. APPROVALS

	Name	Team	Date
--	------	------	------

Prepared by	Vishal Utekar	Open System	06 May 2024
Verified by			
Approved by			