Next Generation Payment Gateway

API Specification (Redirection v1.4)

Distribution of the document

Duplication and distribution of this document without an authorized release is strictly prohibited. In-Solutions Global Ltd. will decide on the number of copies that will be in circulation and the persons with whom the document will be available.

Every person in custody of this document has the responsibility for ensuring its confidentiality. The custodian of the document will also ensure that the document is continually updated with amendments that may be issued from time to time. Any loss or mutilation of the document must be reported promptly to
In-Solutions Global Ltd.

REVISION HISTORY

Sr.No	Ver. No.	Rev. No.	Date of Version	Created / Updated By	Approved By	Section Changed	Description of change
1	01	00	31/Dec/2021	Jay Vyas	Pooja Gowda	Initial Document	
2	01	1	31/10/2022	Sanjana Gawde	Ajay D. Golania	HTTP Post Request Parameter HTTP Post Response Parameter	COFT Tokenization Parameter Changes for RuPay.
3	01	2	01/06/2023	Sanjana Gawde	Ajay D. Golania	Command Parameter Change	COFT Tokenization Parameter Changes for Command.
4	01	3	11/10/2023	Mustafa Siddiqui	Ajay D. Golania	HTTP Post Request Parameters	Added table for the card parameters to be passed based on flow in different Scenario.
5	01	4	15/01/2024	Sanjana Gawde	Ajay D. Golania	HTTP Post Request Parameters	Guest Checkout Parameters added.

TABLE OF CONTENTS

1. INTRODUCTION

Payment Gateway provides merchants a low integration and customized flow-driven solution to integrate their payment-enabled websites and e-commerce applications with the payment networks. It is suitable for most website hosting environments as merchants can integrate payment capabilities into their application without installing or configuring any payments software.

This guide describes how to enable payment to your e-commerce application or online store by using the functionality of the Payment Gateway.

This document is specifically for those merchants who are not PCI compliant and cannot capture card-holder related sensitive information like Card-number, Card Expiry date, Card CVV, etc.

2. TRANSACTION FLOW

2.1. SALE (PURCHASE) TRANSACTION

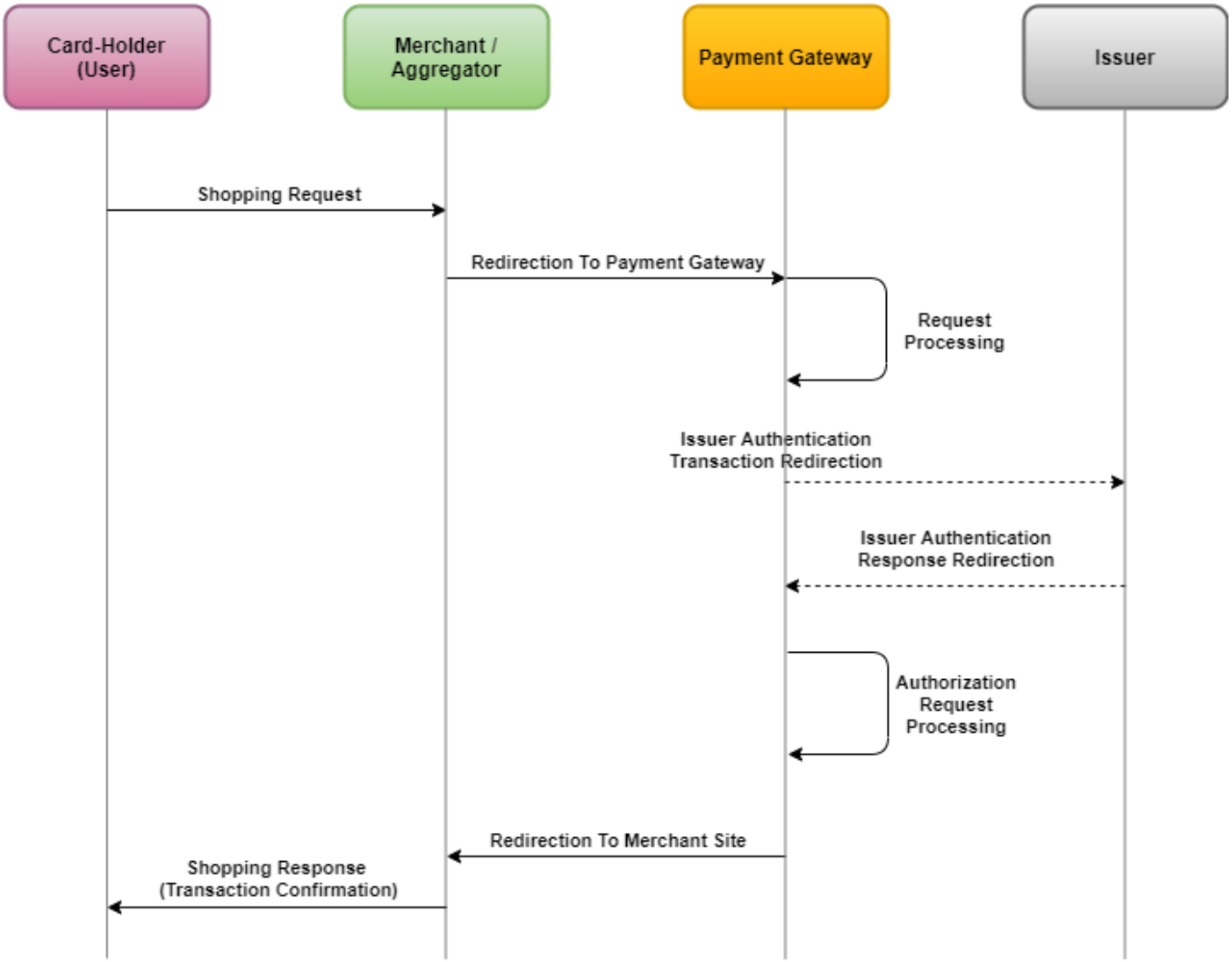

Fig. Sale Transaction

1. When a cardholder initiates a payment request on the merchant website, the merchant redirects card-holder to Payment Gateway with proper request using HTTPS Form Post method.

2. ISGPay Payment Gateway validates and processes the transaction further.

3. Once the transaction will get completed, Payment Gateway will generate appropriate response for merchant and redirect card-holder back to merchant's website on provided Return URL.

4. Based on the response received from Payment Gateway, the merchant will display transaction status to card-holder into their browser.

2.2. SALE STATUS QUERY

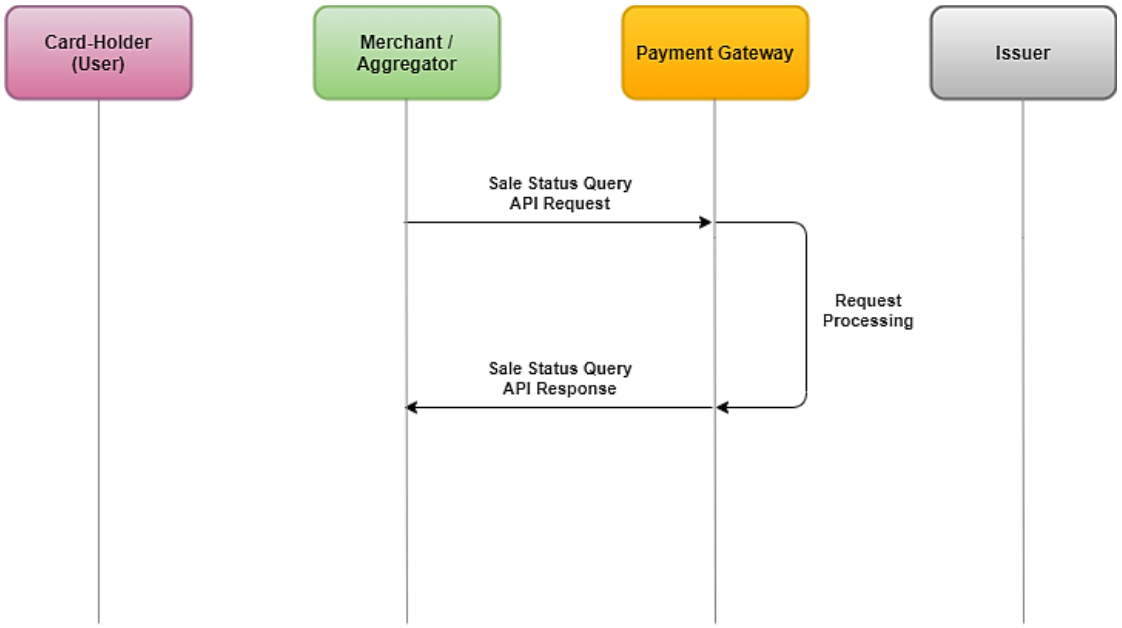

Fig. Sale Status Query API Flow

1. Merchant initiates Sale Status Query API request to Payment gateway to verify sale transaction status at the Payment Gateway level.

2. Payment Gateway accepts status query API request.

3. After successful validation of the request, Payment Gateway fetches the transaction status and generates an API response.

4. Payment Gateway provides the response of transaction on the same channel.

5. Merchant receives the transaction response and process the same at their end.

2.3. REFUND TRANSACTION

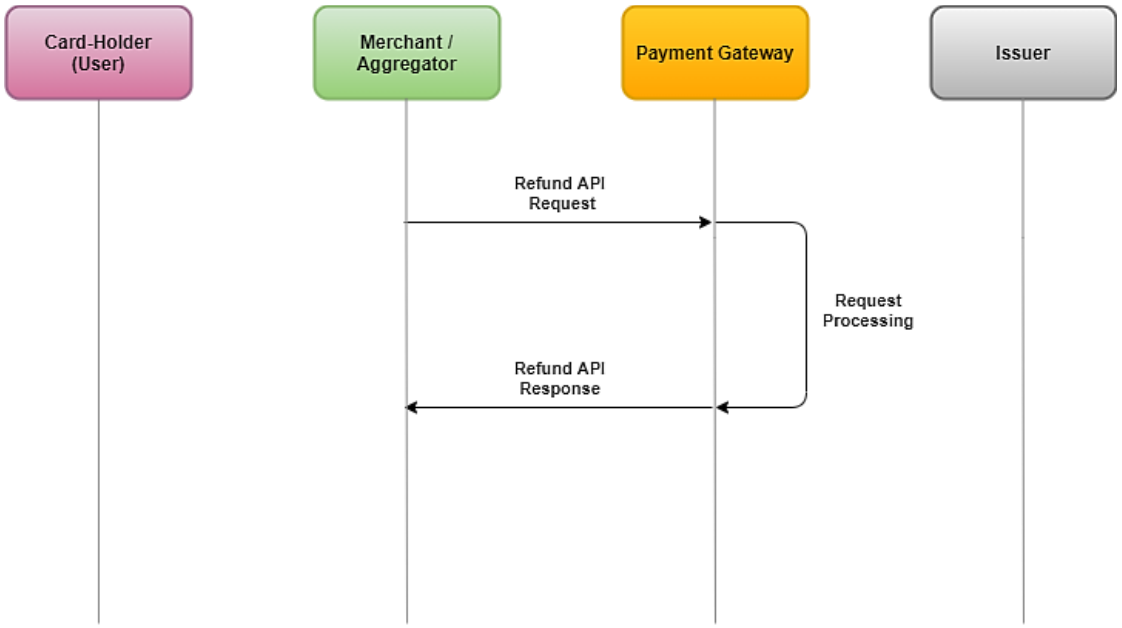

Fig. Refund API Flow

1. Merchant initiate refund request to Payment Gateway to process refund received from card-holder to Payment Gateway.

2. Payment Gateway accepts refund request.

3. After successful validation of the request, Payment Gateway processes the refund and generates an appropriate response.

4. Payment Gateway provides the response of refund transaction on the same channel.

5. Merchant receives the refund transaction response and process the same at their end.

2.4. REFUND STATUS QUERY

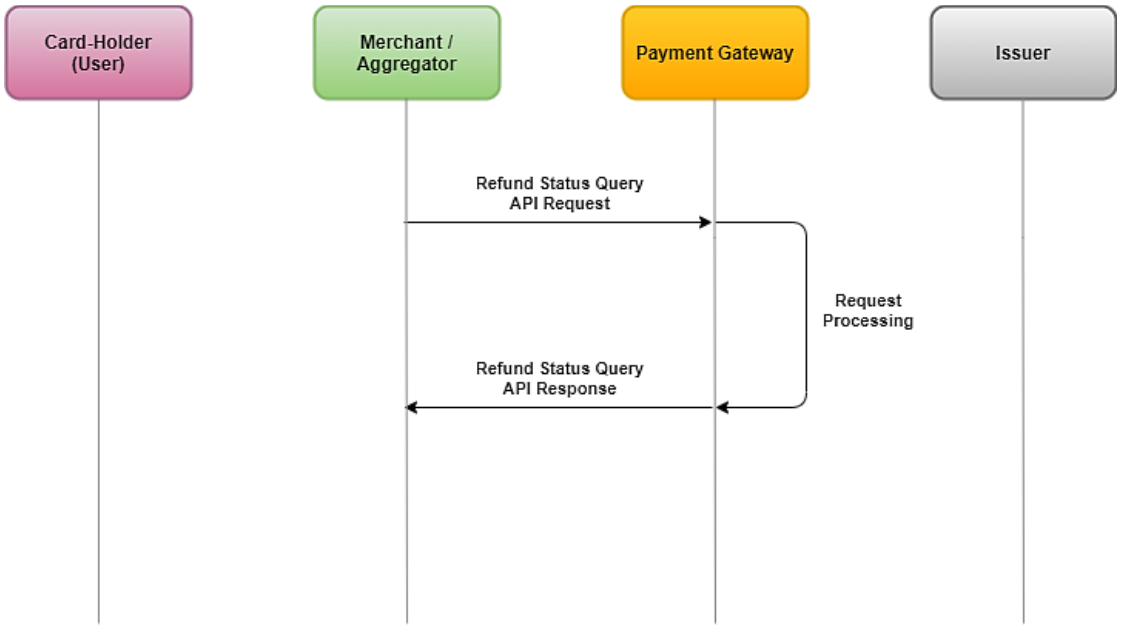

Fig. Refund Status Query API Flow

1. Merchant initiate refund status query API request to Payment Gateway to verify refund transaction status at the Payment Gateway level.

2. Payment Gateway accepts refund status query API request.

3. After successful validation of the request, Payment Gateway fetches the refund transaction status and generates an API response.

4. Payment Gateway provides the response of refund transaction on the same channel.

5. Merchant receives the transaction response and process the same at their end.

2.5. CAPTURE TRANSACTION

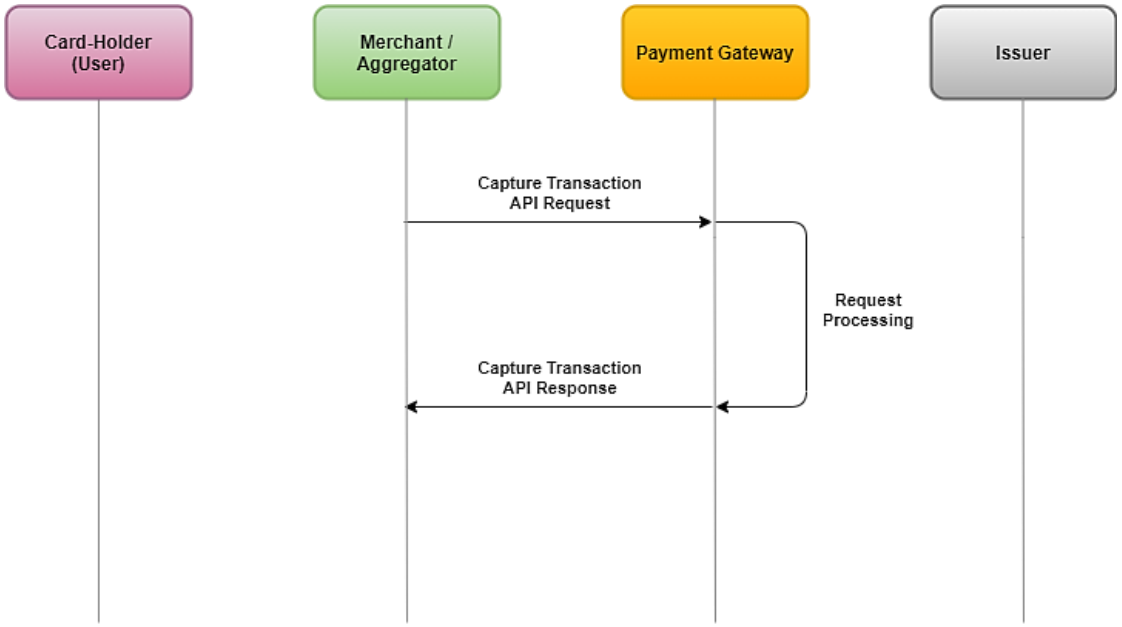

Fig. Capture Transaction API Flow

1. Merchant initiate transaction capture request to the Payment Gateway for receiving funds for Authorized transactions.

2. Payment Gateway accepts capture transaction request.

3. After successful validation of the request, Payment Gateway processes the capture transaction and generates an API response.

4. Payment Gateway provides the response of capturing transactions on the same channel.

5. Merchant receives the transaction response and process the same at their end.

2.6. CALLBACK API (PUSH TRANSACTION)

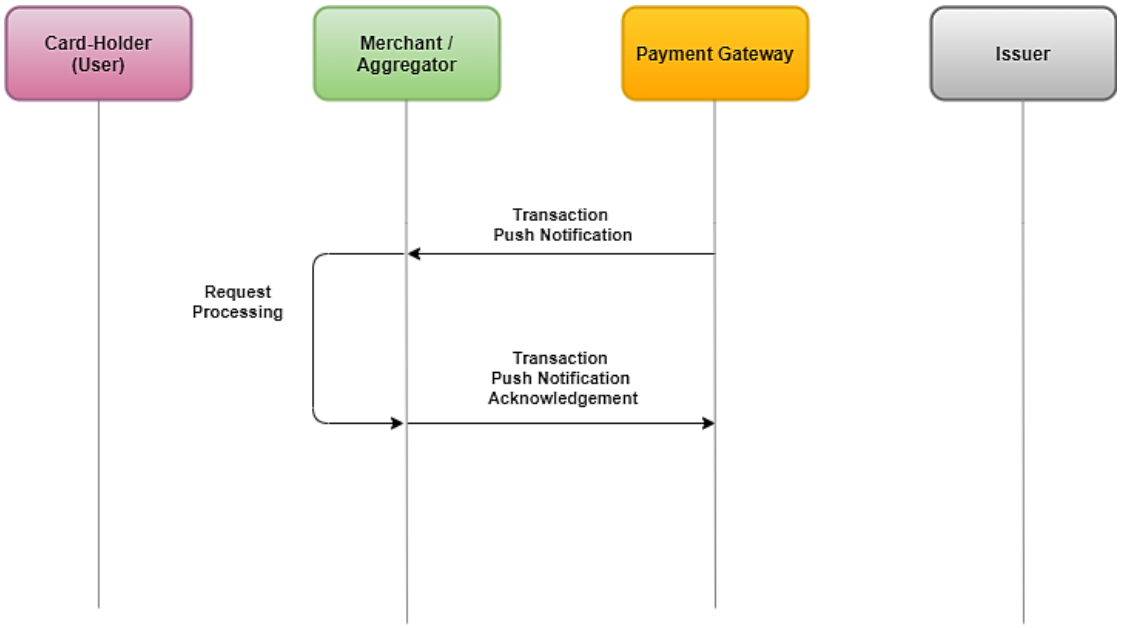

Fig. Transaction Push Notification

1. When Payment Gateway receives transaction response from Acquiring switch, PG initiates transaction response push API if the merchant is registered for Push Transaction Response.

2. Merchant server accepts push transaction request.

3. After successful validation of the request, the Merchant processes the request and generates an API response.

4. Merchant provides acknowledgment to the same channel to Payment Gateway.

5. Payment Gateway receives an acknowledgment from the merchant server.

3. PAYMENT GATEWAY APIs

Payment Gateway API supports HTTP-POST method only with TLS v.1.2 and above. All APIs supports HTTP-Post requests. Other types of requests or HTTP methods are not supported by these APIs.

3.1. SALE TRANSACTION API

For any Sale/Purchase/Auth transaction, the merchant needs to redirect user on this API. This API provides the facility to process Sale/Purchase/Auth transactions. Final transaction response i.e. after transaction processing completion, user will be redirect back to your website with final transaction confirmation response.

Below section provides the details about the request and response parameter for this API.

3.1.1. HTTP Post Request Parameters

1. Main Request Parameters.

Sr.No.	Parameter Name	Data Type	Length	Sample Value
1	MerchantId	Alpha-Numeric	16	100000020000001
2	TerminalId	Alpha-Numeric	8	CG000001
3	BankId	Alpha-Numeric	6	000004
4	EncData	Alpha-Numeric		

Below are the parameters encrypted by using the encryption key at the time of onboarding for EncData parameter data.

EncData must be encrypted in JSON format only. Else, the request will get rejected.

2. EncData Request Parameters.

Sr. No.	Parameter Name	Required/ Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	MCC	Required	Numeric	4	7211
6	PassCode	Required	Alpha-Numeric	8	OXEY8157
7	TxnType	Required	Alpha-Numeric	3-10	Pay
8	Currency	Required	Numeric	3	356
9	Amount	Required	Numeric	1-12	12345
10	OrderInfo	Optional	Alpha-Numeric	1-40	NARUTO00001
11	payOpt	Conditional	Alpha-Numeric	2-10	cc
12	CardNumber	Conditional	Numeric	13-19	5453010000095323
13	ExpiryDate	Conditional	Numeric	6	082025
14	CardSecurityCode	Conditional	Numeric	3	123
15	ReturnURL	Required	Alpha-Numeric	1-255	https://<domain>/IRedirect/merchantResponse.jsp
16	FirstName	Optional	Alpha-Numeric	1-30	Naruto
17	LastName	Optional	Alpha-Numeric	1-30	Uzumaki
18	Street	Optional	Alpha-Numeric	1-30	Hokage Palace
19	City	Optional	Alpha-Numeric	1-30	Konoha
20	State	Optional	Alpha-Numeric	1-30	Hidden Leaf
21	ZIP	Optional	Alpha-Numeric	6	400092

22	Email	Optional	Alpha-Numeric	1-30	naruto@narutoget.com
23	Phone	Optional	Alpha-Numeric	10	9999999999
24	UDF01	Optional	Alpha-Numeric	1-500	UDF01
25	UDF02	Optional	Alpha-Numeric	1-500	UDF02
26	UDF03	Optional	Alpha-Numeric	1-500	UDF03
27	UDF04	Optional	Alpha-Numeric	1-500	UDF04
28	UDF05	Optional	Alpha-Numeric	1-500	UDF05
29	UDF06	Optional	Alpha-Numeric	1-500	UDF06
30	UDF07	Optional	Alpha-Numeric	1-500	UDF07
31	UDF08	Optional	Alpha-Numeric	1-500	UDF08
32	UDF09	Optional	Alpha-Numeric	1-500	UDF09
33	UDF10	Optional	Alpha-Numeric	1-500	UDF10
34	BankCode	Conditional	Alpha-Numeric	1-6	2001
35	chTokenizationConsent	Conditional	Alphabet	1	Y
36	chUserID	Conditional	Alpha-Numeric	1-40	NarutoUzumaki13
37	CardTokenPan	Conditional	Alpha-Numeric	13-19	5564121111106434
38	CardTokenExpiry	Conditional	Numeric	6	082025
39	CardTokenCrypto	Conditional	Alpha-Numeric with	1-100	jdbsbrjeknrkl3io/vndkj=
40	MerchantTRID	Conditional	Alpha-Numeric	1-50	MVTRNO00001
41	CardTokenReferenceNo	Conditional	Alpha-Numeric	1-100	PGTR000000012324
42	PanSource	Conditional	Alphabet	4	KEYE
43	Command	Required	Alpha-Numeric	3-10	Pay Pay+TP
44	AlternateId	Conditional	Numeric		5564121111106434
45	AlternateExpiry	Conditional	Numeric		082025
46	AlternateCrypto	Conditional	Alpha-Numeric with		jdbsbrjeknrkl3io/vndkj=
47	PanAccountReferenceNumber	Conditional	Alpha-Numeric		
48	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

- Note:

1. PanSource Parameter is Mandatory for RuPay Tokenization

2. Command ‘Pay+TP’ is Mandatory when Merchant/Aggregator will be using transaction response for tokenization purpose.

3. AlternateId, AlternateExpiry and AlternateCrypto these fields are only valid for Visa and MasterCard Transaction.

4. PanAccountReferenceNumber field is presently Conditional, it will be mandatory for future use.

- Below table represents the card parameters to be passed based on flow.

Sr. No.	Scenarios	Mandatory Card Related Parameters	Sample Values
1	Card Number Based Transaction	Command CardNumber ExpiryDate CVV	PAY 1234-5678-9012-3456 022033 123
2	Token based transaction (When Merchant/Aggregator is Token Requestor)	Command CardTokenPan CardTokenExpiry CardTokenCrypto	PAY 1234-5678-9012-3456 022033 dsbhfdsjfdshgs==
3	Token based transaction (When Merchant/Aggregator is using ISG's on-behalf service of Token Requestor)	Command CardNumber ExpiryDate CVV chTokenizationConsent chUserID	PAY 1234-5678-9012-3456 022033 123 Y USR12345834
4	Token based sub-sequent transaction (When Merchant/Aggregator is using ISG's on-behalf service of Token Requestor)	Command CardTokenReferenceNo	PAY V.3487294237

3.1.2. HTTP Post Response Parameters

1. Main Response Parameters.

Sr.No.	Parameter Name	Data Type	Length	Sample Value
1	MerchantId	Alpha-Numeric	16	100000020000001
2	TerminalId	Alpha-Numeric	8	CG000001
3	BankId	Alpha-Numeric	6	000004
4	EncData	Alpha-Numeric		

Below are the parameters encrypted by using provided encryption key at the time of onboarding for EncData parameter data.

EncData will be encrypted in JSON format.

2. EncData Response Parameters.

Sr. No.	Parameter Name	Required / Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	MCC	Required	Numeric	4	7211
6	PassCode	Required	Alpha-Numeric	8	OXEY8157
8	Currency	Required	Numeric	3	356
9	Amount	Required	Numeric	1-12	12345
10	OrderInfo	Optional	Alpha-Numeric	1-40	NARUTO00001
11	payOpt	Conditional	Alpha-Numeric	2-10	cc
12	FirstName	Optional	Alpha-Numeric	1-30	Naruto
13	LastName	Optional	Alpha-Numeric	1-30	Uzumaki
14	Street	Optional	Alpha-Numeric	1-30	Hokage Palace
15	City	Optional	Alpha-Numeric	1-30	Konoha
16	State	Optional	Alpha-Numeric	1-30	Hidden Leaf
17	ZIP	Optional	Alpha-Numeric	6	400092
18	Email	Optional	Alpha-Numeric	1-30	naruto@narutoget.com
19	Phone	Optional	Alpha-Numeric	1-30	9999999999
20	UDF01	Optional	Alpha-Numeric	1-500	UDF01
21	UDF02	Optional	Alpha-Numeric	1-500	UDF02
22	UDF03	Optional	Alpha-Numeric	1-500	UDF03
23	UDF04	Optional	Alpha-Numeric	1-500	UDF04
24	UDF05	Optional	Alpha-Numeric	1-500	UDF05
25	UDF06	Optional	Alpha-Numeric	1-500	UDF06
26	UDF07	Optional	Alpha-Numeric	1-500	UDF07
27	UDF08	Optional	Alpha-Numeric	1-500	UDF08
28	UDF09	Optional	Alpha-Numeric	1-500	UDF09
29	UDF10	Optional	Alpha-Numeric	1-500	UDF10
30	MaskedCardNumber	Conditional	Alpha-Numeric	13-19	545301XXXXXX5323
31	ResponseCode	Required	Alpha-Numeric	1-10	00
32	Message	Required	Alpha-Numeric	1-100	Transaction Successful

33	RetRefNo	Conditional	Alpha-Numeric	12	
34	AuthCode	Conditional	Alpha-Numeric	6	
35	chUserID	Conditional	Alpha-Numeric	1-40	NarutoUzumaki13
36	CardTokenReferenceNo	Conditional	Alpha-Numeric	1-100	MVTRNO00001
37	CardTokenResponseCode	Conditional	Alpha-Numeric	1-10	00
38	CardTokenResponseMessage	Conditional	Alpha-Numeric	1-100	Card tokenization in progress
39	UCAP	Conditional	Alpha-Numeric	2	02
40	CAVV	Conditional	Alpha-Numeric	1-100	Gifehfkwqbiufewi==
41	ENROLLED	Conditional	Alpha-Numeric	1	Y
42	AuthStatus	Conditional	Alpha-Numeric	1	A
43	RupayTranId	Conditional	Alpha-Numeric	1-50	10012202210190000000000185164
44	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.2. SALE STATUS QUERY API

Sale Status Query API contains the details about transaction status query API. Whenever a merchant wants to know transaction status at the Payment Gateway end, the merchant can use this API to get the status of a particular transaction.

Below section provides the details about the request and response parameter for this API.

3.2.1. Status Query API Request Parameters

1. Request Parameters.

Sr. No.	Parameter Name	Required/Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	PassCode	Required	Alpha-Numeric	8	OXEY8157
6	TxnType	Required	Alpha-Numeric	3-10	Status
7	MCC	Required	Numeric	4	7211
8	SecureHash	Required	Alpha-Numeric		

3.2.2. Status Query API Response Parameters

1. Response Parameters.

Sr. No.	Parameter Name	Required / Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	PassCode	Required	Alpha-Numeric	8	OXEY8157
6	ResponseCode	Required	Alpha-Numeric	1-10	00
7	Message	Required	Alpha-Numeric		Transaction Successful
8	RetRefNo	Optional	Alpha-Numeric	12	
9	AuthCode	Optional	Alpha-Numeric	6	
10	Amount	Optional	Numeric	1-12	100
11	TxnType	Required	Alpha-Numeric	3-10	Status
13	BatchNo	Conditional	Alpha-Numeric	3-10	102022
14	AuthStatus	Conditional	Alpha-Numeric	1	A
15	Cavv	Conditional	Alpha-Numeric	1-100	Gifehfkwqbiufewi==
16	Enrolled	Conditional	Alpha-Numeric	1	Y
17	Ucap	Conditional	Alpha-Numeric	2	02
18	pgTxnId	Conditional	Alpha-Numeric	2	02
19	UDF01	Optional	Alpha-Numeric	1-500	UDF01
20	UDF02	Optional	Alpha-Numeric	1-500	UDF02
21	UDF03	Optional	Alpha-Numeric	1-500	UDF03
22	UDF04	Optional	Alpha-Numeric	1-500	UDF04
23	UDF05	Optional	Alpha-Numeric	1-500	UDF05
24	UDF06	Optional	Alpha-Numeric	1-500	UDF06
25	UDF07	Optional	Alpha-Numeric	1-500	UDF07
26	UDF08	Optional	Alpha-Numeric	1-500	UDF08
27	UDF09	Optional	Alpha-Numeric	1-500	UDF09
28	UDF10	Optional	Alpha-Numeric	1-500	UDF10

29	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.3. REFUND TRANSACTION API

Refund API contains the details about transaction refunds. Whenever a merchant wants to initiate a refund of a particular sale transaction, the merchant can use this API.

For Auth/Cap type of merchant, the refund will only be processed if there is a capture transaction available for such transaction.

Below section provides the details about the request and response parameter for this API.

3.3.1. Refund API Request Parameters

1. Request Parameters.

Sr. No.	Parameter Name	Required / Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	PassCode	Required	Alpha-Numeric	8	OXEY8157
6	RefundAmount	Required	Numeric	1-12	100
7	RetRefNo	Required	Alpha-Numeric	12	
8	AuthCode	Conditional	Alpha-Numeric	8	
9	RefCancelID	Required	Alpha-Numeric	1-40	CAN- ORD00001
10	TxnType	Required	Alpha-Numeric	3-10	Refund
11	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.3.2. Refund API Response Parameters

1. Response Parameters.

Sr. No.	Parameter Name	Required/ Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	PassCode	Required	Alpha-Numeric	8	OXEY8157
6	RefundAmount	Required	Numeric	1-12	12345
7	Status	Required	Alpha-Numeric	1-10	00
8	Message	Required	Alpha-Numeric		Transaction Successful
9	RetRefNo	Required	Alpha-Numeric	12	
10	AuthCode	Conditional	Alpha-Numeric	8	
11	RefCancelId	Required	Alpha-Numeric	1-40	CAN- ORD00001
12	TxnType	Required	Alpha-Numeric	3-10	Refund
13	BatchNo	Conditional	Alpha-Numeric	3-10	102022
14	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.4. REFUND STATUS QUERY API

Refund Status Query API contains the details about the status of refund transactions. Sometimes, merchant want to know the refund transaction status at Payment Gateway end and to check the status at Payment Gateway, the merchant can use this API to fetch refund status at their end.

Below section provides the details about the request and response parameter for this API.

3.4.1. Refund Status Query API Request Parameters

1. Request Parameters.

Sr. No.	Parameter Name	Required / Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	PassCode	Required	Alpha-Numeric	8	OXEY8157
6	RefCancelId	Required	Alpha-Numeric	1-40	CAN- ORD00001
7	TxnType	Required	Alpha-Numeric	3-10	RefundStatus
8	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.4.2. Refund Status Query Response Parameters

1. Response Parameters.

Sr. No.	Parameter Name	Required/ Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	AccessCode	Required	Alpha-Numeric	8	OXEY8157
6	RefundAmount	Required	Numeric	1-12	12345
7	Status	Required	Alpha-Numeric	1-10	00
8	Message	Required	Alpha-Numeric		Transaction Successful
9	MaskedCardNumber	Required	Alpha-Numeric		
10	RetRefNo	Required	Alpha-Numeric	12	
11	RefCancelId	Required	Alpha-Numeric	1-40	CAN- ORD00001
12	TxnType	Required	Alpha-Numeric	3-10	RefundStatus
13	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.5. CAPTURE API

Capture API contains the details about capturing the transactions.

A pre-authorization is essentially a temporary hold placed by a merchant on a customer's credit card and reserves funds for a future payment transaction. This hold typically lasts about five days, though this depends on your MCC (merchant classification code).

During the hold period, the funds are unavailable to the customer – they won't be able to withdraw it from an ATM or to spend it elsewhere. Although the funds cannot be accessed in their account, no money has been debited in the pre-auth, it is simply reserved. When the time comes to finalize a payment – for example, checking out of a hotel – the funds on hold can then be "captured", meaning they are converted to a charge.

For capturing hold transactions, the merchant can use this API.

Below section provides the details about the request and response parameter for this API.

3.5.1. Capture API Request Parameters

1. Request Parameters.

Sr. No.	Parameter Name	Required/ Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	PassCode	Required	Alpha-Numeric	8	OXEY8157
6	RefCancelId	Required	Alpha-Numeric	1-40	CAN- ORD00001
7	TxnType	Required	Alpha-Numeric	3-10	Capture
8	CaptureAmount	Required	Numeric	1-12	12345
9	RetRefNo	Required	Alpha-Numeric	12	
10	AuthCode	Conditional	Alpha-Numeric	8	
11	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.5.2. Capture API Response Parameters

1. Response Parameters.

Sr. No.	Parameter Name	Required/ Optional	Data Type	Length	Sample Value
1	BankId	Required	Alpha-Numeric	6	000004
2	MerchantId	Required	Alpha-Numeric	16	100000020000001
3	TerminalId	Required	Alpha-Numeric	8	CG000001
4	TxnRefNo	Required	Alpha-Numeric	1-40	ORD00001
5	PassCode	Required	Alpha-Numeric	8	OXEY8157
6	CaptureAmount	Required	Numeric	1-12	12345
7	Status	Required	Alpha-Numeric	1-10	00
8	Message	Required	Alpha-Numeric		Transaction Successful
9	MaskedCardNumber	Required	Alpha-Numeric		
10	RetRefNo	Required	Alpha-Numeric	12	
11	RefCancelId	Required	Alpha-Numeric	1-40	CAN- ORD00001
12	BatchNo	Conditional	Alpha-Numeric	3-10	102022
13	TxnType	Required	Alpha-Numeric	3-10	Capture
14	AuthCode	Conditional	Alpha-Numeric	8	
15	FinalResponse	Conditional	Alpha-Numeric	1-20	
16	SecureHash	Required	Alpha-Numeric		A587436549652BCFAB8768685789CCDA68692BBF

3.6. MERCHANT CALLBACK API (PUSH RESPONSE)

Merchant Call Back API (Push Response) contains the details about transaction push responses from Payment Gateway.

Merchants has to share URL in which ISGPay will post success/failure response similar to return URL which you pass in the “ReturnURL” parameter.

To whitelist call back URL the merchant has to provide their call back URL.

Once data is posted on the call back URL, we receive the response code from the merchant serve for successful communication. If we do not receive the response code from the merchant’s server then we will repost the same data again on call back URL

Reposting will be done a maximum of 3 times in 30 minutes intervals. The callback URL response we will post in JSON. Kindly refer to the below-mentioned sample response format for the same.

Merchant needs to decrypt the data by using the encryption key provided at the time of onboarding.

Below section provides the details about the request and response parameter for this API.

3.6.1. Push Response API Request Parameters

1. Main Request Parameters.

Sr.No.	ParameterName	Data Type	Length	Sample Value
1	MerchantId	Alpha-Numeric	16	100000020000001
2	TerminalId	Alpha-Numeric	8	CG000001
3	BankId	Alpha-Numeric	6	000004
4	EncData	Alpha-Numeric		

3.6.2. Push Response API Response Parameters

1. Response Parameters

Sr.No.	ParameterName	Data Type	Length	Sample Value
1	MerchantId	Alpha-Numeric	16	100000020000001
2	TerminalId	Alpha-Numeric	8	CG000001
3	BankId	Alpha-Numeric	6	000004
4	Acknowledgement	Alpha-Fixed	8	Received

4. SHA-256 SIGNATURE GENERATION

The merchant code creates the SHA-256 Secure Hash value on the Transaction Request data. The Payment Gateway creates another SHA-256 Secure Hash value and sends it back to the merchant in the Transaction Response.

The Secure Hash is a Hex encoded SHA-256 output of a concatenation of all the data parameters. The order that the data parameters are hashed in is extremely important as different transactions contain different data fields so rather than giving the explicit order for each parameter, the order that parameters are hashed in should follow the following rules:

The Secure Hash Secret is always first, thereafter all parameters are concatenated to the secret in natural order of the parameter name. More specifically, the data sort should be in ascending order of the ASCII value of each parameter's name, for example, 'Card' comes before 'card'. Where one string is an exact substring of another, the smaller string should be ordered before the longer, for example, 'Card' should come before 'CardNum'.

Fields must not have any separators between them and must not include any null terminating characters or the like. For example, if the secret is 0F5DD14AE2E38C7EBD8814D29CF6F6F0 and the Transaction Request includes only the following parameters:

Field Name	Example Value
MerchantId	MER123
OrderInfo	Order456
Amount	2995

In ascending alphabetical order, the input to the SHA-256 Secure Hash creation routine would be:

0F5DD14AE2E38C7EBD8814D29CF6F6F02995MER123Order456

This string is then passed through the merchant's SHA-256 Secure Hash generator in the programming language the merchant is using. This output is then converted into hex format (for example, a value of f43c85acfc4e659dcc0f654c553a199797a57cb45a8f5772770271a13fbe287b) and then included in the Transaction Request using the SecureHash field.

The Virtual Payment Client also includes the SecureHash in the Transaction Response so the merchant can check the security of the receipt data. This is performed by first stripping off the SecureHash, and then performing the same steps as creating an SHA-256 Secure Hash for the

Transaction Request, but using the received Transaction Response data fields instead.

The received SecureHash is then compared with the SHA-256 Secure Hash calculated from the Transaction Response data.

If both SHA-256 signatures are the same, the data has not been changed in transit. If they are different, the data needs to be doubled checked.

NOTE : In above table, we have given example using three parameters. Merchant needs to use all the parameters which are going to be part of encData json string for hash generation.

5. AES-256 ENCRYPTION MECHANISM

The request is sent in an encrypted format. All the request parameters along with the Secure Hash are encrypted using the industry-approved AES-256 algorithm. The encrypted value is generated using the request/response parameters along with its corresponding secure hash value.

The merchant code creates the Encrypted value on the Transaction Request data. The Payment Gateway decrypts this value.

The encrypted value is generated creating the string of the bewlo request parameters with the parameters separated by " :: (double colon) ". The parameter name and value are separated by " || (double pipes) ". Fields must not have any separators between them and must not include any null terminating characters.

The parameters may not necessarily be in ascending order of their ASCII value, as in the case for generating Secure Hash.

For example, if the Transaction Request includes only the following parameters:

Field Name	Example Value
MerchantId	100000020000001
TerminalId	CG000001
TxnRefNo	210315162206
BankId	000004

The input string for encryption would be:

MerchantId ||100000020000001:: TerminalId || CG000001:: TxnRefNo ||210315162206:: BankId ||000004

This input string is then encrypted with the AES-256 algorithm using the merchant's encryption key. This encrypted data value is then decrypted at Payment Gateway and on successful decryption, the Transaction Request is processed further.

The Transaction Response from Payment Gateway is also sent in an encrypted format. The Transaction Response will only contain EncData parameter.

NOTE : We have given example in above table. Merchant needs to use parameters listed in respective API's EncData request or response section.

6. STORAGE OF SALT AND ENCRYPTION KEY

You must keep your Secure Hash Secret stored securely. Do not store your secret within the source code of an ASP, JSP, or another website page as it is common for web server vulnerabilities to be discovered where the source code of such pages can be viewed.

You should store your Secure Hash Secret in a secured database, or in a file that is not directly accessible by your web server and has suitable system security permissions.

7. RESPONSE CODES

Sr. No.	Response Code	Message
1	00	Transaction Successful
2	01	Refer to card issuer
3	03	Invalid Merchant Details
4	04	Capture Card Or Hotlisted Card
5	05	Do not honor
6	06	Issuer System Error
7	07	Pickup Card
8	08	Transaction Timed Out
9	10	Partial Approval
10	12	Invalid Transaction
11	13	Invalid Amount
12	14	Invalid Card Number
13	15	Invalid Issuer
14	17	Customer Cancellation
15	19	Re-enter Transaction
16	21	No Action Taken
17	25	Unable To Locate Record In File
18	30	Switch ISO Format Error
19	31	Invalid BIN
20	32	Partial Reversal
21	34	Suspected Fraud
22	3DSF	3DS authentication failed
23	41	Lost Card
24	412	Transaction got declined at issuer.
25	43	Stolen Card
26	51	Insufficient Fund
27	52	No Checking Account
28	53	No Savings Account
29	54	Expired Card
30	55	Invalid PIN
31	57	Transaction Not Permitted To Issuer Or Cardholder
32	58	Transaction Not Permitted To Acquirer Or Merchant
33	59	Suspected Fraud
34	60	Contact Card Acquirer
35	61	Exceeds Withdrawal Amount Limit
36	62	Restricted Card
37	63	Security Violation
38	65	Exceeds Withdrawal Count Limit
39	68	Response Received Late
40	70	Contact Card Issuer
41	71	PIN Not Changed
42	8	Invalid Response From Switch
43	80	Insufficient Fund
44	85	Account Verification Transaction

45	87	Purchase Amount Only, No Cash Back Allowed
46	9	Acquirer Issuer Response Error
47	91	Issuer Unavailable
48	92	Unable to route transaction
49	94	Duplicate transaction Data of STAN or RRN Number
50	96	Issuer System Failure
51	ACCU400	User was inactive.
52	ACCU600	Invalid Data received
53	ACCU700	Duplicate Data posted or Session already expired
54	ACCU800	General Error Encountered
55	BNF	Bin Not Found
56	CAN	Cancel
57	CE	Capture Amount Exceeded
58	CHKBINSIGNNOTMATCHED	Stage 1 Response Digital Signature MissMatched
59	CTO	Network Connect Time Out
60	CTO	Connection Timed Out While Connecting To NPCI.
61	D1	Connection Timed Out
62	D2	Directory Server Didn't Send Response In Specified Time
63	D3	Unable to Connect to Directory Server through Proxy
64	DAUTH	Transaction Declined By Payment Gateway.
65	DE	Switch Insert Failed
66	DOI	Validation Error-Duplicate Order Id Received
67	DTNA	Domestic Transaction not allowed
68	E	The cardholder is not enrolled
69	EMISIGNNOTMATCHED	EMI Response Digital Signature MissMatched
70	FE	Switch ISO Format Error
71	GENOTPSIGNNOTMATCHED	Stage 2 Response Digital Signature MissMatched
72	HNM	Hash Not Matched
73	IAPM	Issuer Authentication Parameter Mismatch
74	IER	Unknown Host
75	IER	Internal Error
76	IER	Transaction could not be processed by Acquiring System
77	IR	InValid Response Parameter from NPCI
78	IR	InValid URL from NPCI
79	IR	Something Went Wrong.
80	ISSHNM	Hash Mismatched with Issuer
81	ISSPM	Parameter Mismatch
82	ISSPMR	Parameters Missing in response
83	IT	Invalid Transaction
84	ITNA	International Transaction not allowed
85	ITO	Late Authorization Request
86	IVR	Invalid Request
87	JFERR	Encrypted Data Is Not In Proper Json Format
88	MECONGMPSW	Improper MPI/Switch Configuration
89	MER001	Merchant Name Is Invalid
90	MNE	Merchant not Enrolled
91	MNE	Merchant Not Enrolled

92	N7	Do Not Honor
93	OTPCAN	User Pressed Cancel Button
94	OTPIER	Something Went Wrong! Please Try Again!
95	OTPPARAMISS08	Auth Value not received from Network
96	PARES001	Validation Error-Blank or null PARes
97	PARES002	Invalid PARes Received
98	PARES002	Invalid PARes Received
99	RAE	Request Amount Exceeded
100	RDF	Duplicate Request Found
101	RE	Refund Amount Exceeded
102	RETRY	You have entered incorrect OTP
103	RNF	Request Not Found
104	RTO	Network Read Time Out
105	RTO	NPCI Didn't send any data in a specified time period.
106	STO	Session Timeout
107	STO01	Transaction Already Authorized or Invalid Authorization Request Data
108	STO02	Transaction authorization is already in progress
109	U	Unable to Verify
110	UC	Unable to connect to a card network
111	VER	Validation Error
112	VER001	MID,TID or BID not sent in request
113	VER002	OrderId was not sent in the request
114	VER003	EncData was not sent in the request
115	VERACCCD01	Validation Error-Blank or null AccessCode
116	VERACCCD02	Validation Error-AccessCode Length Invalid
117	VERACCCD03	Validation Error-AccessCode Contains Invalid Characters
118	VERACCCD04	Validation Error-Access Code MissMatched
119	VERAMT01	Validation Error-Blank or null Amount
120	VERAMT02	Validation Error-Amount Length Exceeds
121	VERAMT03	Validation Error-Amount Contains Non-Numeric Characters
122	VERAMT04	Validation Error-Zero Amount Transaction Not Allowed
123	VERAUTHU01	Validation Error-Blank or null Authentication Response URL
124	VERAUTHU02	Validation Error-Authentication Response URL Length Exceeds
125	VERBRW01	Validation Error-Blank or null Browser Details
126	VERCARD01	Validation Error- CardNumber Not Found In Request
127	VERCARD02	Validation Error- Card Expiry Date Not Found In Request
128	VERCARD03	Validation Error- Card CVV/CVD Not Found In Request
129	VERCARD04	Invalid Card Number
130	VERCARD05	Invalid Card Expiry Date
131	VERCARD06	Invalid Card CVV
132	VERCARDMCNA	MasterCart Network is not allowed for merchant
133	VERCARDRUPNA	RuPay Card is not allowed for merchant
134	VERCARDVNA	Visa Card is not allowed for merchant
135	VERCCNA	Credit Card is not allowed for merchant
136	VERCMD01	Validation Error-Blank or null Command
137	VERCMD02	Validation Error-Invalid Command
138	VERCUR01	Validation Error-Blank or null Currency Code
139	VERCUR02	Validation Error-Amount Length Exceeds
140	VERCUR03	Validation Error-Currency Code Contains Non-Numeric Characters

141	VERCUR04	Currency Not Supported
142	VERCUR04	Currency Not Supported
143	VERKDCEMITT01	Validation Error-Blank or null TxnType
144	VERKDCEMITT02	Validation Error-Invalid TxnType
145	VERKDCEMIET01	Validation Error-Blank or null EmiTenure
146	VERKDCEMIET02	Validation Error-EmiTenure Contains Non-Numeric Characters
147	VERKDCEMI01	Validation Error-Blank or null Product
148	VERKDCEMI02	Validation Error-Product Length Error
149	VERKDCEMI03	Validation Error-Blank or null ProductCategory
150	VERKDCEMI04	Validation Error-ProductCategory Length Error
151	VERKDCEMI05	Validation Error-Blank or null ProductSubCategory
152	VERKDCEMI06	Validation Error-ProductSubCategory Length Error
153	VERKDCEMI07	Validation Error-Blank or null ProdDesc
154	VERKDCEMI08	Validation Error-ProdDesc Length Error
155	VERKDCEMI09	Validation Error-Blank or null ManufacturerName
156	VERKDCEMI10	Validation Error-ManufacturerName Length Error
157	VERKDCEMI11	Validation Error-Blank or null SerialNumber
158	VERKDCEMI12	Validation Error-SerialNumber Length Error
159	VERKDCEMI13	Validation Error-Blank or null Consent
160	VERKDCEMI14	Validation Error-Consent Length Error
161	VERKDCEMI15	Validation Error-Invalid Consent
162	VERKDCEMI16	Authnetication Was Failure
163	VERDCNA	Debit Card is not allowed for merchant
164	VEREMAIL01	Invalid CardHolder Email Id
165	VEREMI01	Validation Error- EMI transaction not allowed!
166	VEREMI02	Validation Error- EmiTenure is Empty or Null
167	VEREMI03	Validation Error- EmiInterestRate is Empty or Null
168	VEREMI04	Validation Error- EmiAmount is Empty or Null
169	VEREMI05	Validation Error- No Plans Available For EmiTenure & EmiInterestRate
170	VEREMI06	Invalid EMI Details
171	VERIP01	Validation Error-Blank or null IP Address
172	VERIP02	Validation Error-ReturnURL Length Too Short
173	VERIP03	Validation Error-IP Address Length Exceeds
174	VERIP04	Validation Error-Invalid IP Address
175	VERMCC01	Validation Error-Blank or null MCC
176	VERMCC02	Validation Error-MCC Length Error
177	VERMCC03	Validation Error-MCC Contains Invalid Characters
178	VERMCC04	Validation Error-MCC MisMatched
179	VERMCCTXN01	Only INR Transaction Is Allowed
180	VERMEID01	Validation Error-Blank or null MerchantId
181	VERMEID02	Validation Error-Merchant Id MissMatched
182	VERMEID03	Validation Error-MerchantId Length Error
183	VERMEID04	Validation Error-MerchantId Contains Invalid Characters
184	VERMOB01	Invalid CardHolder Mobile Number
185	VERMOB02	Invalid phone number
186	VEROINF01	Validation Error-OrderInfo Length Error
187	VEROINF02	Validation Error-OrderInfo Contains Invalid Characters
188	VERORDID01	Validation Error-Blank or null TxnRefNo
189	VERORDID02	Validation Error-Order Id MissMatched

190	VERORDID03	Validation Error-Order Id Length Error
191	VERORDID04	Validation Error-Order Id Contains Invalid Characters
192	VEROTPSIGNNOTMATCHED	Stage Auth Response Digital Signature MissMatched
193	VEROTPSIGNNOTMATCHED	Stage Reversal Response Digital Signature MissMatched
194	VEROTPSIGNNOTMATCHED	Stage 3 Response Digital Signature MissMatched
195	VERPAYOPT	Validation Error- 2Party Payment Option Not Found As Per Standerd
196	VERPAYOPT	Validation Error- Payment Option Not Found As Per Standard
197	VERPGID01	Validation Error-Blank or null PG ID
198	VERPGID02	Validation Error-PG Id not matched with Authentication Request
199	VERTID01	Validation Error-Blank or null TerminalId
200	VERTID02	Validation Error-Terminal Id MissMatched
201	VERTID03	Validation Error-TerminalId Length Error
202	VERTID04	Validation Error-TerminalId Contains Invalid Characters
203	VERUAG01	Validation Error-Blank or null User Agent
204	XY	Expired Card
205	VERPANSC01	Validation Error-Blank or null PanSource
206	VERPANSC02	Validation Failed Invalid PanSource
207	VERALTID01	Blank or Null Alternate Id
208	VERALTID02	Invalid Alternate Id
209	VERALTID03	Blank or Null Alternate Expiry Date
210	VERALTID04	Invalid Alternate Expiry Date
211	VERALTID05	Blank or Null Alternate Crypto
212	VERALTID06	Alternate Crypto Length Exceeds
213	VERALTID02	Invalid Alternate Id
214	VERALTID03	Blank or Null Alternate Expiry Date
215	VERALTID04	Invalid Alternate Expiry Date
216	VERCARDGC	Validation Error : Card Transaction Not Allowed
217	VERCNTDTRAI01	Validation Error : CardNumber or TokenDetails or TokenRefNo or AlternateId not Received in Request
218	VERTDAI03	Validation Error : TokenDetails and AlternateId Received in Request
219	VERCNAI05	Validation Error : CardNumber and AlternateId Received in Request
220	VERTDTRAI01	Validation Error : TokenDetails or TokenRefNo or AlternateId Received in Request Along with Card Number
221	VERCNTRAI01	Validation Error : CardNumber or TokenRefNo or AlternateId Received in Request Along with Card Token Pan
222	VERCNTDAI01	Validation Error : CardNumber or TokenPan or AlternateId Received in Request Along with Card Token Reference No
223	VERCNTDTR01	Validation Error : CardNumber or TokenPan or TokenRefNo Received in Request Along with Alternate ID
224	VERCNCTCRAI05	Validation Error : CardNumber,TokenDetails and TokenRefNo and AlternateId Received in Request

8. API URLs & CREDENTIALS

8.1. Merchant Credentials:

Sr. No.	Parameters	Values
1	Bank ID	<Will be provided during project setup / onboarding>
2	Merchant ID	101000000000781
3	Terminal ID	10100781
4	MCC	5111
5	Pass Code	SVPL4257
6	Secure Secret (SALT)	E59CD2BF6F4D86B5FB3897A680E0DD3E
7	Encryption Key	5EC4A697141C8CE45509EF485EE7D4B1

8.2. Test Card Numbers :

Sr. No.	Network	Card Number	Expiry Date	CVV	OTP
1	MasterCard	5453 0100 0009 5323	01/2025	123	111111
2	Visa	4005 5598 7654 0	01/2025	123	111111
3	RuPay	6071 4898 7654 3212 6074 8299 0000 4938	12/2028	123	123456

8.3. UAT URLs

Sr.No.	API Names	URLs
1	Sale API	https://<domain>/ISGPay-Genius/request.action
2	Sale Status API	https://<domain>/ISGPay-Genius/Status
3	Refund/Capture Transaction	https://<domain>/ISGPay-Genius/RAC
4	Refund Status API	https://<domain>/ISGPay-Genius/RefundStatus

8.4. Production URLs

Sr.No.	API Names	URLs
1	Sale API	https://<domain>/ISGPay-Genius/request.action
2	Sale Status API	https://<domain>/ISGPay-Genius/Status
3	Refund/Capture Transaction	https://<domain>/ISGPay-Genius/RAC
4	Refund Status API	https://<domain>/ISGPay-Genius/RefundStatus

9. PARAMETER GLOSSARY

Sr. No.	Parameter Names	Description
1	Amount	The amount of the transaction, expressed in the smallest currency unit.The amount must not contain any decimal points, thousands separators or currency symbols. For example, Rs. 101.20 is expressed as 10120 This value cannot be negative or zero
2	AuthCode	Authorization Code assigned by Card Issuer upon approval of transaction
3	AuthStatus	Authentication successful evedence
4	BankCode	Issuing Bank Id assigned by Payment System for Net-banking Transactions
5	BankId	Unique ID used for identification of bank
6	BatchNo	Batch number from Payment Gateway
7	CaptureAmount	Amount to be captured in case of Pre-Auth merchant
8	CardNumber	The number of the card used for the transaction .The format of the Card Number is based on the Electronic Commerce Modelling Language (ECML) and , in particular, must not contain white space or formatting characters
9	CardSecurityCode	The Card Security Code (CSC), also known as CVV (Visa), CVC2 (MasterCard) or CID/4DBC (Amex) or CVV2, which is printed, not embossed on the card.It compares the code with the records held in the card issuing institution's database
10	CardTokenCrypto	A Cryptogram is a dynamic, one-time use code for each transaction that accompanies the token. The TAVV cryptogram is a 20-byte Base64-encoded binary value required by network. Note : This field is mandatory for merchants/aggregators who will be processing transaction using card token pan instead of clear card number
11	CardTokenExpiry	Token expiry date provided by Token Service Provider(TSP). Note : This field is mandatory for merchants/aggregators who will be processing transaction using card token pan instead of clear card number.
12	CardTokenPan	A Token replaces sensitive account information, such as the 16-digit account number, with a unique digital identifier. Network stores the relationship between the PAN and token in its secure Token Vault. The token allows payments to be processed without exposing actual account details that could potentially be compromised. Note : This field is mandatory for

		merchants/aggregators who will be processing transaction using card token pan instead of clear card number.
13	CardTokenReferenceNo	The unique reference allocated to the new Token. Serves as a unique identifier for all subsequent queries or management functions relating to this Token. example: "DWSPMC000000000132d72d4fcb 2f4136a0532d3093ff1a45" This value will be provided by Payment Gateway to merchant/aggregator who are/will be using on behalf service of TR.
14	CardTokenResponseCode	Response code for tokenized card whether it was successful or failure.
15	CardTokenResponseMessage	Message for tokenized card whether it was successful or failure
16	CAVV	Authentication successful evedence
17	chTokenizationConsent	Consent taken from card-holder to tokenized his/her card during transaction leg while cardholder goes for save card functionality at Merchant/Aggregator level. Value accepted by PG for this field is “Y” or “N”
18	chUserID	Unique identifier of card-holder at merchant/aggregator end.
19	City	City of customer who placed order
20	Command	Constant value as Pay “Pay+TP” is Mandatory when Merchant/Aggregator will be using transaction response for tokenization purpose.
21	Currency	Different countries have their currency code. For example, • INDIA-356 • USA-840 • KUWAIT-414
22	Email	Email Id of customer who placed order
23	EncData	The encrypted value of request parameters is sent For more details refer “ENCRYPTION” on page 12. Note: The encryption key is provided by the Payment Provider.
24	ENROLLED	Authentication successful evedence
25	ExpiryDate	The expiry date of the card in the format MMYYYY. The value must be expressed as a 6-digit number (integer) with no white space or formatting characters For example, an expiry date of May 2017 is represented as 052017
26	FirstName	First name of customer who placed order
27	LastName	Last name of customer who placed order
28	MaskedCardNumber	Masked card number for reference purpose

29	MCC	MCC(Merchant Category Code) is the Code assigned to business by credit card companies.5974 is used for miscellaneous(different) and specialty retail stores.
30	MerchantId	The unique Merchant Id assigned to a merchant by the Payment Provider. The Merchant ID identifies the merchant account against which settlements will be made
31	MerchantTRID	ID provided by TSP that identifies the Token Requestor.
32	Message	This is a message to indicate what sort of error, if any, the transaction encountered
33	OrderInfo	Unique identification number of customer order
34	PassCode	Authenticates the merchant on the Payment Gateway
35	payOpt	Payment Method for two party transactions: cc-Credit Card dc-Debit Card nb-NetBanking upi - upi wt-Wallet For 3-party merchant, if this parameter is sent then, only respective card option will be displayed on card capture page of Payment Gateway
36	pgTxnId	Unique id generated at Payment Gateway level
37	Phone	Phone number of customer who placed order
38	RefCancelId	Cancellation id of refund transaction
39	RefundAmount	Amount to be refunded
40	ResponseCode	A response code that is generated by the Payment Server to indicate the status of the transaction. A ResponseCode of “00” (Double zero) indicates that the transaction was processed successfully and approved by the acquiring bank.
41	RetRefNo	RetRefNo(Reference Retrieval Number or RRN)is a unique identifier that is passed back to the cardholder for their records if the merchant application does not generate its own receipt number.
42	ReturnURL	URL supplied by the merchant .It is used by the Payment Gateway to redirect the card holder's browser back to the merchant's web site. It must be a fully qualified URL starting with HTTPS:// and if typed into a browser with Internet access, would take the browser to that web page.
43	SecureHash	This is a hash of the fields sent to ensure integrity of the transaction data. For more details refer “Creating a SHA-256 Signature for

		Transactions" on page 11. Note: The secure secret is provided by the Payment Provider.
44	State	State of customer who placed order
45	Status	Transaction status (response code)
46	Street	Street(Address) of customer who placed order
47	TerminalId	Card acceptor terminal identification
48	TxnRefNo	A unique value created by the merchant This identifier will be displayed in the Transaction Search results in the Merchant Web Portal of the Payment Gateway
49	TxnType	Payment Type • Purchase-Pay • Refund-Refund • StatusQuery-Status
50	UCAP	Authentication successful evedence
51	UDF01	User Defined Field
52	UDF02	User Defined Field
53	UDF03	User Defined Field
54	UDF04	User Defined Field
55	UDF05	User Defined Field
56	UDF06	User Defined Field
57	UDF07	User Defined Field
58	UDF08	User Defined Field
59	UDF09	User Defined Field
60	UDF10	User Defined Field
61	ZIP	Postal Code of customer who placed order
62	PanSource	For **RuPay Tokenization** transactions, Merchant needs to send this value in Request there is **fixed** value for this field. PanSource : **KEYE**
63	RupayTranId	A Unique Id that is assigned by NPCI for RuPay Transactions in transaction Authentication Leg.

64	AlternateId	This is conditional filed for Guest Checkout ALT-ID implementation. When merchant/aggregator is directly connected with ALT-ID provider and getting ALT-ID from provider, same value needs to pass in this field.
65	AlternateExpiry	This is conditional filed for Guest Checkout ALT-ID implementation. When merchant/aggregator is directly connected with ALT-ID provider and getting ALT-ID from provider, same value needs to pass in this field.
66	AlternateCrypto	This is conditional filed for Guest Checkout ALT-ID implementation. When merchant/aggregator is directly connected with ALT-ID provider and getting ALT-ID from provider, same value needs to pass in this field.
67	PanAccountReferenceNumber	PanAccountReferenceNumber typically a 14- to 19-digit number that serves as a unique identifier on credit and debit cards as well as other cards that stores value. This field is presently Conditional, it will be mandatory for future use.