TechProcess Payment Services Ltd

Single URL -Integration & Installation Guide

Last updated August 2nd, 2016, Mumbai

“Current Kit Version 5.0”

**Document Control**

|  |  |
| --- | --- |
| Document Title | Single URL Integration and Installation Kit - For Java Framework 1.5 and Higher Versions |
| Document ID |  |
| Document Current Version | CURRENT KIT VERSION 5.0 |
| Effective Date | Aug 02, 2016 |

**Document Revision History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version No. | Version Date | Revised By | Revision  Description | Change Approver | |
| Name | Designation |
| Ver. 4.0 | Sep-2, 2014 | Preeti Jha | Process Described | 'Rajeev Narayanan' | EVP |
| Ver. 5.0 | Aug-02-2016 | Ravi Shankar | Process Described | 'Rajeev Narayanan' | CSIO |

**Document Revision Log For The Project**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Project Version | Change Reference | Changed By |
|  |  |  |  |
|  |  |  |  |

**Document Publication History**

|  |  |
| --- | --- |
| Prepared By  Name – Preeti Jha  Designation – Sr. Business Analyst  (TECHPROCESS ) | Signature - |
| Reviewed By  Name –Rahul Mahapatra  Designation – Chief Manager - Technology  (TECHPROCESS ) | Signature - |

*.*

Contents

[1. Introduction of “The Single URL” Integration & Installation Guide 4](#_Toc411865761)

[2. What is “Single URL- capability”? 4](#_Toc411865762)

[3. Payment -Transaction Flow 5](#_Toc411865763)

[3.1. Flow Description 6](#_Toc411865764)

[4. Technical Prerequisites 7](#_Toc411865765)

[5. Request Type Integration 8](#_Toc411865770)

[Description of Request Types: 9](#_Toc411865771)

[“Transaction Request Types”: 10](#_Toc411865772)

[5.1. “T” Request Type Integration 12](#_Toc411865773)

[5.2. “S” Request Type Integration 13](#_Toc411865781)

[5.3. “O” Request Type Integration 16](#_Toc411865782)

[5.4. Sample O Verification Response for Different Transaction Status For System 18](#_Toc411865783)

[5.5 “Refund Initiation Request Type” 19](#_Toc411865784)

[5.6. “TWC” Request Type Integration 20](#_Toc411865775)

[5.7. “TRC” Request Type Integration 22](#_Toc411865774)

[5.8. “TCC” Request Type Integration 26](#_Toc411865776)

[5.9. “TWI” Request Type Integration 28](#_Toc411865777)

[5.10 “TIC” Request Type Integration (IVR SYSTEM) 31](#_Toc411865778)

[5.11 “TIO” Request Type Integration (IVR SYSTEM) 33](#_Toc411865779)

[6. Server to Server Payment Posting 35](#_Toc411865786)

[7. List of Parameters 37](#_Toc411865788)

# Introduction of “The Single URL” Integration & Installation Guide

The Payment-Gateway technical integration guide helps you, with the detailed steps you need to follow while installing and integrating “The single URL Feature” with your online platform.

**The guide covers below points:**

* Detailed explanation of integration kit and its business benefits.
* Helps you plan the PG integration, implementation, rollout, testing & Go Live with your Team.
* End to end Payment-Transaction flow is briefly explained functionally & technically for your Technical Team.
* Technical prerequisites are listed for the integration of this feature.
* Sample string format are listed your Tech Team.
* Parameters are well explained for building data exchange logic between PG platform and Merchant’s platform.

# What is “Single URL- capability”?

The Single URL capability provides a **secured** data exchange mechanism between “TechProcess PG Platform” and “Merchant’s online platform over internet.

|  |
| --- |
| **Secured Data Exchange – Figure -1.0** |

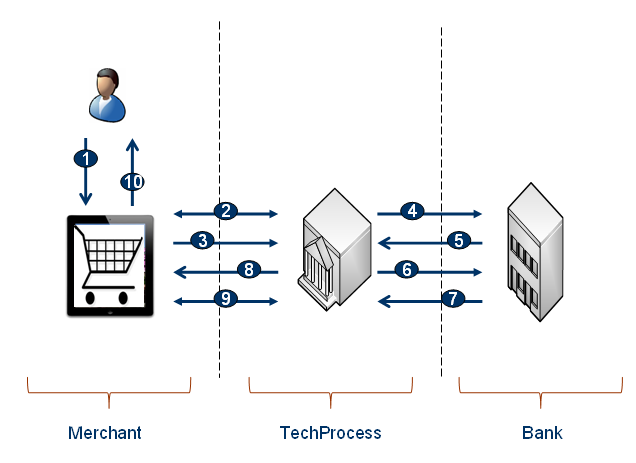
When two Financial Applications are connected, the database connection strings, passwords authenticity and authorization are required careful consideration therefore TechProcess has created a secured data exchange capability, termed as “The Single URL” capability.

The supreme safeguarded methods have been utilized that provides secured environment to exchange sensitive data between “TechProcess PG Platform” and “Merchant’s online platform over internet.

**Other than the Secured Environment features the “Single URL Capability” has following other Benefits also:**

* Merchant not need to configure multiple URLs to integrate with TechProcess PG Platform, as only one URL can technically handle all the request types, that are exchanged during transaction (payment) being made.
* Minimized development & implementation effort at Merchant’s end.
* An encryption technology as **AES encryption** is done for better data security during data exchange between merchant and TECHPROCESS system.
* User friendly bank selection page with minimum clicks for the end User.
* The user friendly browser page provides the user experience such as speedy information display, faster browser & faster data loading that minimizes the user log off probability that helps to increasex Higher Success Ratio.
* The “Card capturing page” can be integrated with either Merchant’s end, TECHPROCESS end or Bank ends as per your requirement.

# Payment -Transaction Flow



**Payment Transaction Flow – Figure -2.0**

# Flow Description

|  |  |
| --- | --- |
| **Steps** | **Description** |
| **Step 1:** | As described in the above **Payment Transaction Flow – Figure -2.0,** Customer comes onto the merchant website, selects the goods/services that he intends to purchase, he adds the goods to the shopping cart. Customer now moves to the checkout process. |
| **Step 2:** | Merchant will send request to TechProcess with the payment details. The request parameters are mentioned below in ***List of Parameter (section 10).***  TechProcess will validate the request and store the payment details in system.  TechProcess will generate token number for the request and send URL in response.  Sample URL:  For UAT:-  Web-ServiceURL:- 1. <https://www.tekprocess.co.in/PaymentGateway/services/TransactionDetailsNew>  Redirection URL:-  <https://www.tekprocess.co.in/PaymentGateway/txnreq.pg?id=03903acc-9e0a-4b9b-ae86-7661a353aa65>  Production:-  Web-Service URL  <https://www.tpsl-india.in/PaymentGateway/services/TransactionDetailsNew>  Redirection URL  https://www.tpsl-india.in/PaymentGateway/txnreq.pg?id=03903acc-9e0a-4b9b-ae86-7661a353aa65 |
| **Step 3:** | Merchant will redirect the customer to the URL received from Techprocess.(This will be a Browser to Browser communication).  If payment-mode(Credit/Debit/Net banking/wallet) selection is at Techprocess end-:  Bank selection page will be displayed to customer and customer can select the appropriate payment mode and proceed further.  If payment-mode(Credit/Debit/Net banking/wallet) selection is at Merchant end-:  Customer will be redirected to the respective bank’s page. |
| **Step 4:** | TechProcess will redirect the customer to the bank.  Customer will enter his login credentials and process the transaction |
| **Step 5:** | Bank will debit customer’s account and provide status to TechProcess. |
| **Step 6 & 7:** | TechProcess will perform online verification and update the status in system. |
| **Step 8:** | TechProcess will forward response to the merchant with payment details. |
| **Step 9:** | Merchant will validate the checksum and send the request for dual verification. |
| **Step 10:** | Merchant notifies cardholder of transaction response for Card Gateway transactions. |

# Technical Prerequisites

|  |  |
| --- | --- |
| For single URL integration Merchant & Techprocess needs to ensure below mentioned facilitation:  Java Prerequisites | With technical Prerequisites |

* The Merchant’s web application should be in JAVA(version 1.4 and above) Environment.
* Port to be open at both (Merchant and Techprocess) ends
* MID (MerchantCode) , this will be shared by Techprocess
* key and iv pair for encryption and decryption , this will be shared by Techprocess
* Java Integration Kit(.jar file to be shared by Techprocess)

.Net Prerequisites

* Port to be open at both (Merchant and Techprocess) ends
* MID(MerchantCode), this will be shared by Techprocess
* key and iv pair for encryption and decryption , this will be shared by Techprocess
* DotNetIntegrationKit.DLL
* Error Property file
* Log File
* Adding Key Pair in web.Config

**Installation Steps for .Net clients**

* 1. Need to refer the provided **DLL** by TECHPROCESS from integration kit.
  2. Please make sure to use only, the provided **key and iv pair**.
  3. To know the error details, please refer the **error property** file which is part of integration kit.
  4. To maintain log information, please create the **“Log”** folder at following location **“C:\PropertyFile\”.**
  5. Add following entries in **web.config** file -

|  |
| --- |
| <appSettings>  <add key="ErrorFile" value="D:\\PropertyFile\\ErrorMessage.property"/>  <add key="LogFilePath" value="D:\\PropertyFile\\Log\\"/>  <add key="TPSLService" value="https://www.tekprocess.co.in/PaymentGateway/services/TransactionDetailsNew"/>  </appSettings> |

PHP Prerequisites

* Merchant should be consume WebService.
* Port to be open at both (Merchant and TECHPROCESS ) ends
* MID(MerchantCode), this will be shared by TECHPROCESS
* key and iv pair for encryption and decryption , this will be shared by TECHPROCESS

Other Technology Prerequisites

* Merchant should be consume WebService.
* Port to be open at both (Merchant and TECHPROCESS ) ends
* MID(MerchantCode), this will be shared by TECHPROCESS
* key and iv pair for encryption and decryption , this will be shared by TECHPROCESS



# Request Type Integration

There are several “Request Types” available in the “Single URL Feature” that can be configured by your Technology Team to process the transaction.

|  |
| --- |
|  |

**Secured Data Exchange via Request Type – Figure -2.0**

Your system or online platform can trigger “Request Type” to TECHPROCESS PG Platform. On the basis of verification in the backend, the result (Response) will be displayed for the user.

# Standard Request Types:

|  |  |  |
| --- | --- | --- |
| **Transaction Request Types** | | |
| **Request Type** | **Request Type Description** | **Rational** |
| T | Payload is for normal Transaction processing. | Applicable for:  1.       Net-banking Transactions. 2.       Wallets/Cash Cards Transactions. 3.       Card transactions (If card details are not captured  at Merchant’s end). |
|
|
| S | Real-time Verification Request | 1. Validate all Success transactions, received in real time response (Request Type – T). 2. Success Status returned in Request Type – S, should be considered for further processing at merchant’s end. |
|
|
| O | Offline Verification Request  (To be initiated only after 30 minute of transaction initiation) | 1. To verify the transaction status for all Pending Transactions in real time. |
|
| R | Refund Initiation Request | 1. Initiating a refund transactions for which fund are already settled with merchant. |
|

# Additional Request Types:

|  |  |  |
| --- | --- | --- |
| **Request Type** | **Request Type Description** | **Rational** |
| **TWC** | **T**ransaction **W**ith **C**ard | Applicable for-:   1. Card transactions where card details are captured at Merchant end. 2. Card need not to be saved. |
| **TRC** | **T**ransaction & **R**egistration with **C**ard | Applicable for-:   1. Card transactions where card details are captured at Merchant end & Merchant wants to save the customer’s card details in Techprocess card vaulting system. |
| **TCC** | **T**ransaction with **C**ardID & **C**VV | Applicable for-:   1. Repeat customers whose card details are already vaulted in Techprocess card vaulting system & the corresponding reference number (CardID) is already available with the merchant. |
| **TWI** | **T**ransaction **W**ith **I**MPS | Applicable for-:   1. IMPS transactions where IMPS data (MMID, Mobile No., OTP) details are captured at Merchant end. |
| **TIC** | **T**ransaction *for* **I**VR with **C**ard**.** | Applicable for-:   1. Merchants who wants to initiate transactions through IVR |
| **TIO** | **T**ransaction *for* **I**VR with **O**TP | Applicable for-:   1. Merchants who want to initiate transactions through IVR along with OTP. |

# “Transaction Request Types”:

# “T” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“T”** |
| Description | Merchant’s requirement is to allow end customer only pay the payment and not opt for standing instructions or register themselves for SI, neither opt for InstaPay feature where InstaPay allows end customer to pay recurring payment without entering all the card details except CVV number. The business requirement is to only allow customer to make payment where Techprocess provides the card capturing page that will be hosted by Techprocess. |
| Flow Details | 1) Use the sample code to get the redirection url.  2) Redirect customer’s browser to the received url. |
| Event /Trigger of Request Type | On a click of pay button the payment request gets triggered. |
| Sample Request Format | **Sample Java Code-:Please refer to this attached document for details.**    **Sample DotNet Code: Please refer to this attached document for details.**    **Sample PHP Code: Please refer to this attached document for details.**    Any Other format (for No-Kit Integration)-:  Rqst\_type=T|rqst\_kit\_vrsn=1.0.1|tpsl\_clnt\_cd=T2551|clnt\_txn\_ref=TXN0001|clnt\_rqst\_meta={email:myname@domain.com}|  rqst\_amnt=12.00|rqst\_crncy=INR|rtrn\_url=https://tpslvksrv6046/LoginModule/Test.jsp|  s2s\_url=https://tpslvksrv6046/LoginModule/Test.jsp|rqst\_rqst\_dtls=Test\_120.0\_0.0|  clnt\_dt\_tm=14-03-2014|tpsl\_bank\_cd=470|hash=8cf71e0eb3f7273113b7e73e0db671d8 |
| Sample Response Format | If Request Type=”T” Response   |  |  |  | | --- | --- | --- | | Response Parameters | Description | Sample Value | | msg | resposne Payload in encrypted format | VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D | | tpsl\_mrct\_cd | Merchant Code | T2658 |   Example :  msg=VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D&tpsl\_mrct\_cd=T2658  **For Java Client**  Sample java code*:* ***Please refer to attached document for details.(Refer previous attached java Sample code)***  **For .Net client :**  Sample DotNet Code***: Please refer to attached document for details.(Refer previous attached .Net Sample code)***  **For PHP clients:**  Sample PHP Code*:* ***Please refer to attached document for details.***  *(Refer previous attached PHP sample code)*  **Final Response to Merchants :Post decryption**  For Success:  txn\_status=0300|txn\_msg=success|txn\_err\_msg=NA|clnt\_txn\_ref=TPSL210915453|tpsl\_bank\_cd=470|tpsl\_txn\_id=117203569|txn\_amt=2.00|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}{email:sunil.sahu@techproces.co.in}{mob:9856987456}|tpsl\_txn\_time=12-11-2014 16:58:24|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=c9831306-23e8-4d54-ac18-b7faa8edb73e|hash=d924c4bdc8b8e542453345bb2b719318b26219e9  For Failed :  txn\_status=0399|txn\_msg=failed|txn\_err\_msg=failure reason|clnt\_txn\_ref=TPSL210915453|tpsl\_bank\_cd=470|tpsl\_txn\_id=117203569|txn\_amt=2.00|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}{email:sunil.sahu@techproces.co.in}{mob:9856987456}|tpsl\_txn\_time=12-11-2014 16:58:24|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=c9831306-23e8-4d54-ac18-b7faa8edb73e|hash=d924c4bdc8b8e542453345bb2b719318b26219e9 |
| **Parsing Response message** | Refer txn\_status code to find the meaning of the status code. |

# “S” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“S”** |
| Description | This is a mandatory service call to get the final transaction status for all real-time success transactions.  Merchant needs to call this request type for each and every success response received on realtime. |
| Event /Trigger of Request Type | Backend – Server to Server Communication |
| Sample Request Format | **Sample Java Code:**    **Sample DotNet Code:** |
| Sample Response Format | **If Request Type=”S” Response**  txn\_status=0300|txn\_msg=SUCCESS|txn\_err\_msg=Normal SUCCESS Transaction|clnt\_txn\_ref=TPSL209177648|tpsl\_bank\_cd=9980|tpsl\_txn\_id=228587|txn\_amt=1.00|tpsl\_txn\_time=14-11-2014 10:03:25|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=a4e3e263-bea5-4f63-ac04-e7a6822e03cb |
| List of parameters For Real Time Transaction verification Payload – (Request Type S) | **Conditional**: clnt\_txn\_ref and tpsl\_txn\_time or tpsl\_txn\_id should be provided  The request parameters are mentioned in ***List of Parameter (section 10).*** |

# “O” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“O”** |
| Description | In a scenario where final success or fail status is not received from TECHPROCESS to the Merchant then, Merchant can request status update though “O” request type anytime after realtime. |
| Event /Trigger of Request Type | Backend – Server to Server Communication |
| Sample Request Format | **Sample Java Code:**    **Sample DotNet Code:** |
| Sample Response Format | **If Request Type=”O” Response**  **FOR NO REFUND TRANSACTIONS**  txn\_status=0300|txn\_msg=SUCCESS|txn\_err\_msg=Success Test Bank Transaction.|clnt\_txn\_ref=TPSL939584727|tpsl\_bank\_cd=470|tpsl\_txn\_id=165291964|txn\_amt=2.00|tpsl\_txn\_time=08-07-2015 14:15:41|tpsl\_rfnd\_id=NA|bal\_amt=2.00|REFUND\_DETAILS=NA|rqst\_token=41f717f6-c4fc-4497-9d2d-3e5a6c7fe43c  **Partial refund -:**  txn\_status=0300|txn\_msg=SUCCESS|txn\_err\_msg=Success Test Bank Transaction.|clnt\_txn\_ref=TPSL939584727|tpsl\_bank\_cd=470|tpsl\_txn\_id=165291964|txn\_amt=2.00|tpsl\_txn\_time=08-07-2015 14:15:41|tpsl\_rfnd\_id=1187861|bal\_amt=1.00| REFUND\_DETAILS={SR\_NO:1~REF\_TXN\_ID:1187861~REF\_TXN\_AMT:1.00~REF\_TXN\_DATE:07-08-2015~STATUS:INITIATED} |rqst\_token=41f717f6-c4fc-4497-9d2d-3e5a6c7fe43c           Transaction status will be success with refund details will be appended in response.  **Full refund -:**  txn\_status=0400|txn\_msg=REFUND|txn\_err\_msg=Success Test Bank Transaction.|clnt\_txn\_ref=TPSL939584727|tpsl\_bank\_cd=470|tpsl\_txn\_id=165291964|txn\_amt=2.00|tpsl\_txn\_time=08-07-2015 14:15:41|tpsl\_rfnd\_id=1187861, 1187862|bal\_amt=0.00|REFUND\_DETAILS={SR\_NO:1~REF\_TXN\_ID:1187861~REF\_TXN\_AMT:1.00~REF\_TXN\_DATE:07-08-2015~STATUS:INITIATED},{SR\_NO:2~REF\_TXN\_ID:1187862~REF\_TXN\_AMT:1.00~REF\_TXN\_DATE:07-08-2015~STATUS:INITIATED}|rqst\_token=  Transaction status will be returned as REFUND, with refund details will be appended in response.  In both the case this service will returns details of all the refunded transactions.  REFUND\_DETAILS={SR\_NO:1~REF\_TXN\_ID:1187861~REF\_TXN\_AMT:1.00~REF\_TXN\_DATE:07-08-2015~STATUS:INITIATED},{SR\_NO:2~REF\_TXN\_ID:1187862~REF\_TXN\_AMT:1.00~REF\_TXN\_DATE:07-08-2015~STATUS:INITIATED} |
| List of parameters For Real Time Transaction verification Payload – (Request Type S) | **Conditional**: clnt\_txn\_ref and tpsl\_txn\_time or tpsl\_txn\_id should be provided  The request parameters are mentioned in [***List of Parameter***](#Paramerter)***.*** |

# Sample O type (Verification) Response for Different Transaction Status For System

---------------------------------------- For Success Transaction (T) -----------------------------------------

txn\_status=0300|txn\_msg=SUCCESS|txn\_err\_msg=Success Test Bank Transaction.|clnt\_txn\_ref=2358|tpsl\_bank\_cd=470|tpsl\_txn\_id=140813616|txn\_amt=10.00|tpsl\_txn\_time=09-02-2015 16:30:32|tpsl\_rfnd\_id=NA|bal\_amt=NA|base\_fee=10.00|tpsl\_charges =0.00|service\_tax =0.00|rqst\_token=27384ed2-4e0c-46b8-b729-aa50204473cb

---------------------------------------- For User Aborted Transaction (U) ----------------------------------

txn\_status=0395|txn\_msg=User Aborte|txn\_err\_msg=PENDING|clnt\_txn\_ref=6911|tpsl\_bank\_cd=470|tpsl\_txn\_id=134206249|txn\_amt=10.00|tpsl\_txn\_time=05-02-2015 16:23:59|tpsl\_rfnd\_id=NA|bal\_amt=NA|base\_fee=10.00|tpsl\_charges =0.00|service\_tax =0.00|rqst\_token=ead42300-c03e-43ed-94dc-ff3883108b6d

---------------------------------------- For Pending Transaction (P) -----------------------------------------

txn\_status=0396|txn\_msg=AWAITED|txn\_err\_msg=PENDING|clnt\_txn\_ref=2760|tpsl\_bank\_cd=470|tpsl\_txn\_id=117214981|txn\_amt=1.00|tpsl\_txn\_time=30-12-2014 12:01:30|tpsl\_rfnd\_id=NA|bal\_amt=NA|base\_fee=1.00|tpsl\_charges =0.00|service\_tax =0.00|rqst\_token=50651a5e-e517-4b7e-8916-de21dba7cf75

---------------------------------------- For Aborted Transaction (D) -----------------------------------------

txn\_status=0397|txn\_msg=ABORTED|txn\_err\_msg=PENDING|clnt\_txn\_ref=2699|tpsl\_bank\_cd=470|tpsl\_txn\_id=117214763|txn\_amt=1.00|tpsl\_txn\_time=29-12-2014 14:56:50|tpsl\_rfnd\_id=NA|bal\_amt=NA|base\_fee=1.00|tpsl\_charges =0.00|service\_tax =0.00|rqst\_token=1be8b2b6-87b8-432a-992c-bbcc3c6528a9

---------------------------------------- For Failed Transaction (F) -------------------------------------------

txn\_status=0399|txn\_msg=FAILED|txn\_err\_msg=Failed Test Bank Transaction. Transaction Cancelled by User|clnt\_txn\_ref=2356|tpsl\_bank\_cd=470|tpsl\_txn\_id=140813598|txn\_amt=10.00|tpsl\_txn\_time=09-02-2015 15:56:21|tpsl\_rfnd\_id=NA|bal\_amt=NA|base\_fee=10.00|tpsl\_charges =0.00|service\_tax =0.00|rqst\_token=4f209fcd-c5b1-4e7e-879f-ce4cd00a9aa1

# “Refund Initiation Request Type”

# “R” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“R”** |
| Description | Merchant’s requirement is to be able to mark refund manually from Admin module . However the request type “R” should be sent through API for **Techprocess** to process the refund. |
| Event /Trigger of Request Type | Backend – Server to Server Communication |
| Sample Request Format | Create the object of TransactionRequestBean class and set the required value in the bean object. **If Request Type is “R”, refer to list of parameters of “R” request Type for setting parameter values in TransactionRequestBean.**  **Sample Java Code:**      **Sample DotNet Code:** |
| Sample Response Format | **If Request Type=”R” Response**  txn\_status=0400|txn\_msg=Refund Initiated|txn\_err\_msg=|clnt\_txn\_ref=NA|tpsl\_bank\_cd=470|tpsl\_txn\_id=71047288|txn\_amt=100|tpsl\_txn\_time=05032014 124902687|tpsl\_rfnd\_id=543272|bal\_amt=0|rqst\_token= |
| List of parameters For Real Time Transaction verification Payload – (Request Type S) | **Conditional**: clnt\_txn\_ref and tpsl\_txn\_time or tpsl\_txn\_id should be provided.  The request parameters are mentioned in List of Parameter (section 10). |
| For Refund Transaction Payload    For Request Type = “R” : Make a request to refund a transaction. | Create the object of TransactionRequestBean class and set the required value in the bean object. **If Request Type is “R”, refer to refer to list of parameters of “R” for setting parameter values in TransactionRequestBean.** Call the objTransactionRequestBean.getTransactionToken() method. This method return the below mention response.  **Example :**  Res\_String= txn\_status=0400|txn\_msg=Refund Initiated|txn\_err\_msg=|clnt\_txn\_ref=14032014150942|tpsl\_bank\_cd=470|tpsl\_txn\_id=71048731|txn\_amt=120.00|tpsl\_txn\_time=14-03-2014 16:13:12|tpsl\_rfnd\_id=543339|bal\_amt=45.44|rqst\_token= |

# “TWC” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“TWC”** |
| Description | Merchant’s requirement is to allow End Customer to make payment by entering the card details where card capturing page is provided by the Merchant. |
| Event /Trigger of Request Type | On a click of pay button the payment request gets triggered. |
| Sample Request Format | **Sample Java Code:**      **Sample DotNet Code:**    **Request data format**:  rqst\_type=TWC|rqst\_kit\_vrsn=1.0.3|tpsl\_clnt\_cd=T2658|clnt\_txn\_ref=TXN0001|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}|rqst\_amnt=2.00|rqst\_crncy=INR|rtrn\_url=https://tpslvksrv6046/TWCTest/Responsepayload.jsp|s2s\_url=https://tpslvksrv6046/LoginModule/Test.jsp|rqst\_rqst\_dtls=Test\_2.0\_0.0|clnt\_dt\_tm=11-03-2014|tpsl\_bank\_cd=9980|card\_details=fbd440061a1b866333e4756e1137c37ca5b84569919b60d19f6e86f33efaee6511ea060747780f8e1b7e55e2b0e0e69880cfb5e9a2bc593d4f9b019fd9f9ed2a076d9abb9f9f1fcf64631d17974a30b1d3fd01c3ff1fc2918801ad0f0834f8e4336c51d3003ba3067159d9b243a445deaa4f07e60661c66dbf22ac01bf7b2adffee17c979ba01837bfabbc9b3a0b689b6f8b58e705c4375921de09af1a7c0057c03c255c847411a50d6383f951b27f61|hash=033cd6718532a848fae444db26fbba1d8a1d4820 |
| Sample Response Format | **If Request Type=”TWC” Response**  txn\_status=0300|txn\_msg=success|txn\_err\_msg=NA|clnt\_txn\_ref=TPSL1117741786|tpsl\_bank\_cd=9980|tpsl\_txn\_id=228583|txn\_amt=1.00|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}|tpsl\_txn\_time=14-11-2014 09:42:39|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=a69a7a4e-b494-4eae-8827-4d95093745e8|hash=77f7303dc01b421615f66e60bf126f6d06c03068 |
| List of parameters For Real Time Transaction verification Payload – (Request Type TWC) | For Transaction Payload – (Request Type TWC)  The request parameters are mentioned in List of Parameter (section 10). |
| **Steps to get response data** | **Response = msg=VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D&tpsl\_mrct\_cd=T2658**  **After getting the response from PG to do the below mention step.**   1. **Create the object of TransactionResponseBean Class.** 2. **Set all the bean value as define below.**   **beanObj.setIv(iv.getBytes());**  **beanObj.setKey(key.getBytes());**  **In below method merchant needs to set the msg parameter value**  **beanObj.setResponsePayload(msg); //set the pg response message**  **Call the beanObj.getResponsePayload() method.**  **This method return the below**  **mention response.**  txn\_status=0300|txn\_msg=success|txn\_err\_msg=NA|clnt\_txn\_ref=TPSL1117741786|tpsl\_bank\_cd=9980|tpsl\_txn\_id=228583|txn\_amt=1.00|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8mar2014~Payment~forpayment}|tpsl\_txn\_time=14-11-2014 09:42:39|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=a69a7a4e-b494-4eae-8827-4d95093745e8|hash=77f7303dc01b421615f66e60bf126f6d06c03068 |

# “TRC” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“TRC”** |
| Description | Merchant’s requirement is to allow End Customer to make payment by entering the card details where card capturing page is provided by the Merchant. Also Merchant wishes to store card information at Merchant’s database to register the card.  Note\*\**The card registration is required for the 1st time payment so that the customer can pay using only CVV number for the subsequent transaction. However for the subsequent transaction the Merchant needs to send “TCC” Request Type.* |
| Event /Trigger of Request Type | On a click of pay button the payment request gets triggered. |
| Sample Request Format | **Request data format:**  **Sample Java Code:**    rqst\_type=TRC|rqst\_kit\_vrsn=1.0.3|tpsl\_clnt\_cd=T2658|clnt\_txn\_ref=TXN0001|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}|rqst\_amnt=2.00|rqst\_crncy=INR|rtrn\_url=https://tpslvksrv6046/TWCTest/Responsepayload.jsp|s2s\_url=https://tpslvksrv6046/LoginModule/Test.jsp|rqst\_rqst\_dtls=Test\_2.0\_0.0|clnt\_dt\_tm=11-03-2014|tpsl\_bank\_cd=9980|card\_details=fbd440061a1b866333e4756e1137c37ca5b84569919b60d19f6e86f33efaee6511ea060747780f8e1b7e55e2b0e0e69880cfb5e9a2bc593d4f9b019fd9f9ed2a076d9abb9f9f1fcf64631d17974a30b1d3fd01c3ff1fc2918801ad0f0834f8e4336c51d3003ba3067159d9b243a445deaa4f07e60661c66dbf22ac01bf7b2adffee17c979ba01837bfabbc9b3a0b689b6f8b58e705c4375921de09af1a7c0057c03c255c847411a50d6383f951b27f61|hash=033cd6718532a848fae444db26fbba1d8a1d4820 |
| Sample Response Format | **If Request Type=”TRC”**  **Response**  txn\_status=0300|txn\_msg=success|txn\_err\_msg=NA|clnt\_txn\_ref=TPSL399698281|tpsl\_bank\_cd=10140|tpsl\_txn\_id=117203564|txn\_amt=2.00  |clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}{email:sunil.sahu@techproces.co.in}{mob:9856987456}|  tpsl\_txn\_time=12-11-2014 16:40:54|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=6308eff9-87db-40f1-8a17-0097c4264818|card\_id=4077|  BankTransactionID=114111201389206|alias\_name=VISA\_DBT\_\*\*\*\*9847|hash=d9cf87c453fbc49f32536db6097d9e80cc7d3397 |
| List of parameters For Real Time Transaction verification Payload – (Request Type TRC) | For Transaction Payload – (Request Type TRC)  The request parameters are mentioned in List of Parameter (section 10). |
| For Transaction & Registration with Card Payload    For Request Type = “TRC” : Make a request to initiate a transaction along with card registration while card capture is at merchant end. | Create the object of TransactionRequestBean class and set the required value in the bean object. **If Request Type is “TRC”, refer to refer to list of parameters of “TRC” for setting parameter values in TransactionRequestBean.**  **Sample Java Code:**  **Example:** TransactionRequestBean objTransactionRequestBean=new TransactionRequestBean();  objTransactionRequestBean.setStrRequestType("TRC");  objTransactionRequestBean.setStrMerchantCode("T2663");  objTransactionRequestBean.setMerchantTxnRefNumber("");  objTransactionRequestBean.setStrAmount("120.00");  objTransactionRequestBean.setStrCurrencyCode("INR");  objTransactionRequestBean.setStrITC("email:myname@domain.com");  objTransactionRequestBean.setStrReturnURL("https://tpslvksrv6046/MerchantIntegrationClient/Responsepayload.jsp ");  objTransactionRequestBean.setStrShoppingCartDetails("Test\_120.0\_0.0");  objTransactionRequestBean.setTxnDate("24-12-2013");  objTransactionRequestBean.setStrBankCode("470");  objTransactionRequestBean.setWebServiceLocator("https://www.tpsl- india.in/PaymentGateway/services/TransactionDetailsNew");  objTransactionRequestBean.setStrTPSLTxnID("71047882");  objTransactionRequestBean.setStrMobileNumber("8451053257");  objTransactionRequestBean.setKey(key.getBytes());  objTransactionRequestBean.setIv(iv.getBytes());  objTransactionRequestBean.setStrCustomerName("xyz");  objTransactionRequestBean.setStrEmail("myname@domain.com");  objTransactionRequestBean.setStrMMID("");  objTransactionRequestBean.setStrOTP("");  objTransactionRequestBean.setCardID("");  objTransactionRequestBean.setCustID("19872627");  objTransactionRequestBean.setCardName("dheeraj");  objTransactionRequestBean.setCardNo("1245142514251425");  objTransactionRequestBean.setCardCVV("125");  objTransactionRequestBean.setCardExpMM("05");  objTransactionRequestBean.setCardExpYY("2014");  objTransactionRequestBean.setStrTimeOut(“1000”) this should be in seconds  Call the objTransactionRequestBean.getTransactionToken () method.  This method will return  Two possible outputs.   1. Will get the transaction URL with Unique token ID.   **Example:**  https://www.tekprocess.co.in/PaymentGateway/txnreq.pg?id=8bdb7304-e6cd-4b08-8168-aa51c438e42b   1. If any validation fail will get only error code. (Starts with ERROR)   **Sample DotNet Code:** |
| Steps to get response data | Response = msg=VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D&tpsl\_mrct\_cd=T2658  After getting the response from PG to do the below mention step.   1. Create the object of TransactionResponseBean Class. 2. Set all the bean value as define below.   beanObj.setIv(iv.getBytes());  beanObj.setKey(key.getBytes());  **In below method merchant needs to set the msg parameter value**  beanObj.setResponsePayload(msg); //set the pg response message  **Call** the beanObj.getResponsePayload() method. This method return the below  mention response.  txn\_status=0300|txn\_msg=success|txn\_err\_msg=|clnt\_txn\_ref=7421002|tpsl\_bank\_cd=470|tpsl\_txn\_id=71047298|txn\_amt=100.00|tpsl\_txn\_time=05-03-2014 23:13:12|card\_ID=12345|Alias\_name=VISA\_DBT\_\*\*\*\*\_1234|BankTransactionId =1234567|rqst\_token= |

# “TCC” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“TCC”** |
| Description | Assuming the card is registered using “TRC” request type at merchant’s end, now the Merchant’s requirement is to allow End Customer to make payment by entering the only CVV number against the register card to pay subsequent payment.  Note\*\**The CVV will be captured on Merchant’s portal.* |
| Event /Trigger of Request Type | On a click of pay button the payment request gets triggered. |
| Sample Request Format | **Sample Java code :**    **Request Format: Payload generation at Merchant End:**  Final Payload to be sent to token generation system with ~(Tilda) appended and MID added at the end :-    encAes(rqst\_type=TCC|rqst\_kit\_vrsn=1.0.1|tpsl\_clnt\_cd=T2551|clnt\_txn\_ref=TXN0001|clnt\_rqst\_meta={email:myname@domain.com}|  rqst\_amnt=12.00|rqst\_crncy=INR|rtrn\_url=https://tpslvksrv6046/LoginModule/Test.jsp|s2s\_url=https://tpslvksrv6046/LoginModule/Test.jsp|rqst\_rqst\_dtls=Test\_120.0\_0.0~Test\_120.0\_0.0|clnt\_dt\_tm=14-03-2014|tpsl\_bank\_cd=470|card\_details= 2dd03a3c9fbfb552f139c9801dace20880391cbf3e5002978fbf05753074ab1c837279863e17dc4fd4448c2eeb61eaf6|Card\_ID=123456|T2856~**encAes =AES Encryption**    Note: The encrypted response will have merchant code appended at the end. |
| Sample Response Format | **If Request Type=”TCC” Response**  txn\_status=0300|txn\_msg=success|txn\_err\_msg=NA|clnt\_txn\_ref=TPSL1755422387|tpsl\_bank\_cd=9980|tpsl\_txn\_id=228582|txn\_amt=1.00|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}{mob:9856987456}|tpsl\_txn\_time=14-11-2014 09:38:57|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=65fe6900-db52-4635-a584-3065dbcbf09b|card\_id=1108|BankTransactionID=2014112574211|hash=d7baacbfe190ec26e6743005c6750691b4c88a3f |
| List of parameters For Real Time Transaction verification Payload – (Request Type TCC) | For Transaction Payload – (Request Type TCC)  The request parameters are mentioned in List of Parameter (section 10). |
| For Transaction with CardID & CVV Payload    For Request Type = “TCC”: Make a request to initiate transaction against Card ID. | Create the object of TransactionRequestBean class and set the required value in the bean object. **If Request Type is “TCC”, refer to refer to list of parameters of “TCC” for setting parameter values in TransactionRequestBean.**  **Sample Java Code:**  **Sample DotNet Code:** |
| **Steps to get response data** | |  |  |  | | --- | --- | --- | | Response Parameters | Description | Sample Value | | msg | resposne Payload in encrypted format | VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D | | tpsl\_mrct\_cd | Merchant Code | T2658 |   Final Response :  txn\_status=0300|txn\_msg=success|txn\_err\_msg=NA|clnt\_txn\_ref=TPSL1755422387|tpsl\_bank\_cd=9980|tpsl\_txn\_id=228582|txn\_amt=1.00|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}{mob:9856987456}|tpsl\_txn\_time=14-11-2014 09:38:57|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=65fe6900-db52-4635-a584-3065dbcbf09b|card\_id=1108|BankTransactionID=2014112574211|hash=d7baacbfe190ec26e6743005c6750691b4c88a3f |

# “TWI” Request Type Integration

|  |  |
| --- | --- |
| Request Type | **“TWI”** |
| Description | Merchant’s requirement is to allow end customer to make a payment using IMPS as a payment option. End customer will enter IMPS details such as MMID, Mobile number & OTP at Merchant’s website |
| Event /Trigger of Request Type | On a click of pay button the payment request gets triggered. |
| Sample Request Format | Request data format:  rqst\_type=TWI|rqst\_kit\_vrsn=1.0.3|tpsl\_clnt\_cd=T2658|clnt\_txn\_ref=TXN0001|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}|rqst\_amnt=2.00|rqst\_crncy=INR|rtrn\_url=https://tpslvksrv6046/TWITest/Responsepayload.jsp|s2s\_url=https://tpslvksrv6046/LoginModule/Test.jsp|rqst\_rqst\_dtls=Test\_2.0\_0.0|clnt\_dt\_tm=11-03-2014|tpsl\_bank\_cd=9980|imps\_details=fbd440061a1b866333e4756e1137c37ca5b84569919b60d19f6e86f33efaee6511ea060747780f8e1b7e55e2b0e0e69880cfb5e9a2bc593d4f9b019fd9f9ed2a076d9abb9f9f1fcf64631d17974a30b1d3fd01c3ff1fc2918801ad0f0834f8e4336c51d3003ba3067159d9b243a445deaa4f07e60661c66dbf22ac01bf7b2adffee17c979ba01837bfabbc9b3a0b689b6f8b58e705c4375921de09af1a7c0057c03c255c847411a50d6383f951b27f61|hash=033cd6718532a848fae444db26fbba1d8a1d4820 |
| Sample Response Format | **Sample Response**  txn\_status=0300|txn\_msg=success|txn\_err\_msg=|clnt\_txn\_ref=7421002|tpsl\_bank\_cd=470|tpsl\_txn\_id=71047298|txn\_amt=100.00|tpsl\_txn\_time=05-03-2014 23:13:12|BankTransactionID=12345| sha1-hash |
| List of parameters For Real Time Transaction verification Payload – (Request Type TWI) | The request parameters are mentioned in List of Parameter (section 10). |
|  | **If Request Type is “TWI”, refer to refer to list of parameters of “TWI” for setting parameter values in TransactionRequestBean.**  **Sample Java Code:**  TransactionRequestBean objTransactionRequestBean=new TransactionRequestBean();  objTransactionRequestBean.setStrRequestType("TWI");  objTransactionRequestBean.setStrMerchantCode("T2663");  objTransactionRequestBean.setMerchantTxnRefNumber("TXN0001");  objTransactionRequestBean.setStrAmount("120.00");  objTransactionRequestBean.setStrCurrencyCode("INR");  objTransactionRequestBean.setStrITC("email:myname@domain.com");  objTransactionRequestBean.setStrReturnURL("https://tpslvksrv6046/MerchantIntegrationClient/Responsepayload.jsp ");  objTransactionRequestBean.setStrShoppingCartDetails("Test\_120.0\_0.0");  objTransactionRequestBean.setTxnDate("24-12-2013");  objTransactionRequestBean.setStrBankCode("470");  objTransactionRequestBean.setWebServiceLocator("https://www.tpsl- india.in/PaymentGateway/services/TransactionDetailsNew");  objTransactionRequestBean.setStrTPSLTxnID(“”);  objTransactionRequestBean.setKey(key.getBytes());  objTransactionRequestBean.setIv(iv.getBytes());  objTransactionRequestBean.setStrCustomerName("xyz");  objTransactionRequestBean.setStrEmail("myname@domain.com");  objTransactionRequestBean.setStrMobileNumber("8451053257");  objTransactionRequestBean.setStrMMID("1231221");  objTransactionRequestBean.setStrOTP("123123");  objTransactionRequestBean.setStrTimeOut(“1000”) this should be in seconds    **Note:** Use key and iv shared by TECHPROCESS .  Call the objTransactionRequestBean.getTransactionToken() method.  This method will return two possible output.  1 If any validation fail will get only error code. (Starts with ERROR)  2 txn\_status=0300|txn\_msg=success|txn\_err\_msg=|clnt\_txn\_ref=7421002|tpsl\_bank\_cd=470|tpsl\_txn\_id=71047298|txn\_amt=100.00|tpsl\_txn\_time=05-03-2014 23:13:12|BankTransactionID=12345| sha1-hash  **Sample DotNet Code:** |
| **Steps to get response data** | **Response = msg=VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D&tpsl\_mrct\_cd=T2658**  **After getting the response from PG to do the below mention step.**   1. **Create the object of TransactionResponseBean Class.** 2. **Set all the bean value as define below.**   **beanObj.setIv(iv.getBytes());**  **beanObj.setKey(key.getBytes());**  **In below method merchant needs to set the msg parameter value**  **beanObj.setResponsePayload(msg); //set the pg response message**  **Call the beanObj.getResponsePayload() method.**  **This method return the below**  **mention response.**  txn\_status=0300|txn\_msg=success|txn\_err\_msg=|clnt\_txn\_ref=7421002|tpsl\_bank\_cd=470|tpsl\_txn\_id=71047298|txn\_amt=100.00|tpsl\_txn\_time=05-03-2014 23:13:12|BankTransactionID=12345| sha1-hash |

# “TIC” Request Type Integration (IVR SYSTEM)

|  |  |
| --- | --- |
| Request Type | **“TIC”** |
| Description | Merchant’s requirement is to allow End Customer to make payment through IVR System. |
| Event /Trigger of Request Type | On customer request for payment Merchant’s IVR system will initiate this request |
| Sample Request Format | **Request data format**:  rqst\_type=TIC|rqst\_kit\_vrsn=1.0.3|tpsl\_clnt\_cd=T2658|clnt\_txn\_ref=TXN0001|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}|rqst\_amnt=2.00|rqst\_crncy=INR|rtrn\_url=https://tpslvksrv6046/TWCTest/Responsepayload.jsp|s2s\_url=https://tpslvksrv6046/LoginModule/Test.jsp|rqst\_rqst\_dtls=Test\_2.0\_0.0|clnt\_dt\_tm=11-03-2014|tpsl\_bank\_cd=9980|card\_details=fbd440061a1b866333e4756e1137c37ca5b84569919b60d19f6e86f33efaee6511ea060747780f8e1b7e55e2b0e0e69880cfb5e9a2bc593d4f9b019fd9f9ed2a076d9abb9f9f1fcf64631d17974a30b1d3fd01c3ff1fc2918801ad0f0834f8e4336c51d3003ba3067159d9b243a445deaa4f07e60661c66dbf22ac01bf7b2adffee17c979ba01837bfabbc9b3a0b689b6f8b58e705c4375921de09af1a7c0057c03c255c847411a50d6383f951b27f61|hash=033cd6718532a848fae444db26fbba1d8a1d4820 |
| Sample Response Format | **If Request Type=”TIC” Response**  token  Example C495B3E7-79A0-413C-B82A-1632920CB244  For any error it will be starts with ERROR |
| **Details of TIC** | For Transaction Payload – (Request Type TIC)  The request parameters are mentioned in List of Parameter (section 10). |
|  | **If Request Type is “TIC”, refer to refer to list of parameters of “TIC” for setting parameter values in TransactionRequestBean.**  **Sample Java Code:**  **Example:**  sTransactionRequestBean objTransactionRequestBean=new TransactionRequestBean();  objTransactionRequestBean.setStrRequestType("TIC");  objTransactionRequestBean.setStrMerchantCode("T2663");  objTransactionRequestBean.setMerchantTxnRefNumber("TXN0001");  objTransactionRequestBean.setStrAmount("120.00");  objTransactionRequestBean.setStrCurrencyCode("INR");  objTransactionRequestBean.setStrITC("email:myname@domain.com");  objTransactionRequestBean.setStrReturnURL("https://tpslvksrv6046/MerchantIntegrationClient/Responsepayload.jsp ");  objTransactionRequestBean.setStrShoppingCartDetails("Test\_120.0\_0.0");  objTransactionRequestBean.setTxnDate("24-12-2013");  objTransactionRequestBean.setStrBankCode("470");  objTransactionRequestBean.setWebServiceLocator("https://www.tpsl- india.in/PaymentGateway/services/TransactionDetailsNew");  objTransactionRequestBean.setStrTPSLTxnID(“”);  objTransactionRequestBean.setStrMobileNumber("8451053257");  objTransactionRequestBean.setKey(key.getBytes());  objTransactionRequestBean.setIv(iv.getBytes());  objTransactionRequestBean.setStrCustomerName("xyz");  objTransactionRequestBean.setStrEmail("myname@domain.com");  objTransactionRequestBean.setCustID(“1234”);  objTransactionRequestBean.setCardNo(“1111123123121111”);  objTransactionRequestBean.setCardName(“Rajeev Narayanan”);  objTransactionRequestBean.setCardExpMM(“11”);  objTransactionRequestBean.setCardExpYY(“2017”);  objTransactionRequestBean.setCardCVV(“111”);  objTransactionRequestBean.setStrTimeOut(“1000”) this should be in seconds  objTransactionRequestBean.setAccountNo("32567891042"); this should be numeric  **Note:** Use key and iv shared by **Techprocess**.  Call the objTransactionRequestBean.getTransactionToken() method.  This method will return  two possible output.   1. Will get the transaction URL with Unique token ID.   **Example:**  C495B3E7-79A0-413C-B82A-1632920CB244   1. If any validation fail will get only error code. (Starts with ERROR) |

# “TIO” Request Type Integration (IVR SYSTEM)

|  |  |
| --- | --- |
| Request Type | **“TIO”** |
| Description | Merchant’s requirement is to allow End Customer to make payment through IVR System. |
| Event /Trigger of Request Type | On customer request for payment Merchant’s IVR system will initiate this request with OTP for authorization. |
| Sample Request Format | **Request data format**:  rqst\_type=TIO|rqst\_kit\_vrsn=1.0.6|tpsl\_clnt\_cd=T2490|tpsl\_txn\_id=9d453ae0-c711-4115-9ce6-37b9472cc1f2|otp=123123|hash=109ab1d64aab489af4f553532bcc71b7ca4dbd8c |
| Sample Response Format | **If Request Type=”TIO” Response**  txn\_status=0300|txn\_msg=success|txn\_err\_msg=|clnt\_txn\_ref=TXN0001|tpsl\_bank\_cd=10590|tpsl\_txn\_id=228501|txn\_amt=1.00|clnt\_rqst\_meta={itc:email:myname@domain.com}{email:myname@domain.com}{mob:9888888888}{custname:xyz}|tpsl\_txn\_time=10-11-2014 09:32:54|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=8a5719f2-d8d6-4b6b-8f21-b82b2d179958|alias\_name=NA|card\_id=|BankTransactionID=20141110093200054228501|hash=e2fa11edc451d979c7c03f665b894d0b69fca107 |
| **Details of TIC** | For Transaction Payload – (Request Type TIO)  The request parameters are mentioned in List of Parameter (section 10). |
|  | **If Request Type is “TIO”, refer to refer to list of parameters of “TIC” for setting parameter values in TransactionRequestBean.**  **Sample Java Code:**  **Example:**  TransactionRequestBean objTransactionRequestBean=new TransactionRequestBean();  objTransactionRequestBean.setStrRequestType("TIO");  objTransactionRequestBean.setWebServiceLocator("https://www.tpsl- india.in/PaymentGateway/services/TransactionDetailsNew");  objTransactionRequestBean.setStrTPSLTxnID(token);  objTransactionRequestBean.setOtp(“123456”)  objTransactionRequestBean.setStrMerchantCode("T2490");  objTransactionRequestBean.setKey(key.getBytes());  objTransactionRequestBean.setIv(iv.getBytes());  objTransactionRequestBean.setStrTimeOut(“1000”) this should be in seconds  **Note:** Use key and iv shared by TPSL. |
| Steps to get response data | Response = msg=VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D&tpsl\_mrct\_cd=T2658  After getting the response from PG to do the below mention step.   1. Create the object of TransactionResponseBean Class . 2. Set all the bean value as define below.   beanObj.setIv(iv.getBytes());  beanObj.setKey(key.getBytes());  In below method merchant needs to set the msg parameter value  beanObj.setResponsePayload(msg); //set the pg response message  **Call** the beanObj.getResponsePayload() method. This method return the below  mention response.  txn\_status=0300|txn\_msg=success|txn\_err\_msg=|clnt\_txn\_ref=TXN0001|tpsl\_bank\_cd=10590|tpsl\_txn\_id=228501|txn\_amt=1.00|clnt\_rqst\_meta={itc:email:myname@domain.com}{email:myname@domain.com}{mob:9820596973}{custname:xyz}|tpsl\_txn\_time=10-11-2014 09:32:54|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=8a5719f2-d8d6-4b6b-8f21-b82b2d179958|alias\_name=NA|card\_id=|BankTransactionID=20141110093200054228501|hash=e2fa11edc451d979c7c03f665b894d0b69fca107 |

# 

# Server to Server Payment Posting

# Techprocess PG System will push the transaction response for every successful transaction to the pre-defined merchant URL in secured server to server communication.

* This Response from **Techprocess** PG system to Merchant will be same as browser to browser “T” type transaction response. Hence, merchant has to ensure that the transaction response is updated only once in its system.
* As this is a Payment posting, **Techprocess** PG system expects an acknowledgment from Merchant server on receipt and processing of data. Merchant has to send an acknowledge of the response in the below format

**Format for acknowledgement to be received from the merchant server:**

**MRCT\_TXN\_ID|PG\_TXN\_ID|STATUS**

**Note:** Inthe following scenarios, **Techprocess** PG will try to resend the transaction response for 3 times.

1. If **Techprocess** PG system does not receive acknowledgement status 1,
2. If the connectivity is not established in between **Techprocess** PG system and Merchant system.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Steps to get response data | Response Parameter Names and Desc :   |  |  |  | | --- | --- | --- | | Response Parameters | Description | Sample Value | | msg | response Payload in encrypted format | VsLXhgOWN2ZopVLnoncP93KVhgHHE%2BpTb5OHrSIh2p%2BSAlgE9NYwF%2F7sKFDqRh%2FNjAmNXumeJP89%0D%0A2oAjgGotnbyCfGdzYS5EYX1j732fUJWJ6plrboJ0v0QCWHhXBA3UBZ33Nbf3VEjOm8xhxnv8PMnd%0D%0AIVpRf%2BQXrSvLRh%2F%2FueNz6PS49RMHjHJmcW6T5PmJpvAv9jkHVCksTFQqFh9izVNZv43jj4vjdr%2Fz%0D%0A1iOU5%2FU0g0uKpZ0hNqzhDXeR0f1OOgcuJ51XsJdZVk9qGUJkoneH8rRR9WKw6h6QoqYSxfq6NoGb%0D%0AiKSvanTo2HbqPeVfOz%2BOYJEIj5aTm5Ju9kk1t2OxFdcVD8dwT6cnzd6tbBd8xHj6aIwPF2hQ%2BGAu%0D%0AJBwMy9g8Z4MOoVHlIRCWgR0f75yBKUQhPsOBxrM231Yy%2F9iPRAV5y4olNL6pE4wYdosseXaAWznh%0D%0ACp9Il6DA38KAnvnMKmHvSo%2BAU4cLrOd9zdORsU177VhnnwzQBJptOgKxh4o1mmRQavMrUmTOKd2y%0D%0Ac0uZmIi8r2yBCebbJpyC594%3D | | tpsl\_mrct\_cd | Merchant Code | T2658 | | clnt\_txn\_ref | Merchant Transaction Reference no | TXN1234 |   Note : Merchant needs to capture above parameters using “get” method  Response  After getting the response from PG to do the below mention step.   1. Create the object of TransactionResponseBean Class . 2. Set all the bean value as defined below   beanObj.setIv(iv.getBytes()); //Techprocess shared IV value  beanObj.setKey(key.getBytes());//Techprocess shared KEY value  beanObj.setResponsePayload(msg); //request.getParameter(“msg”) value  **Call** the beanObj.getResponsePayload() method. This method return the below  mention response.  txn\_status=0300|txn\_msg=success|txn\_err\_msg=NA|clnt\_txn\_ref=TPSL210915453|tpsl\_bank\_cd=470|tpsl\_txn\_id=117203569|txn\_amt=2.00|clnt\_rqst\_meta={itc:NIC~TXN0001~122333~rt14154~8 mar 2014~Payment~forpayment}{email:email@techproces.co.in}{mob:9856987456}|tpsl\_txn\_time=12-11-2014 16:58:24|tpsl\_rfnd\_id=NA|bal\_amt=NA|rqst\_token=c9831306-23e8-4d54-ac18-b7faa8edb73e|hash=d924c4bdc8b8e542453345bb2b719318b26219e9 |
| Steps to send Acknowledgement | On Successful processing of above data Merchant server should write response on below format:  **MRCT\_TXN\_ID|PG\_TXN\_ID|STATUS**  MRCT\_TXN\_ID :Merchant Transaction Reference Number  PG\_TXN\_ID :TPSL Transaction id  STATUS :status of the transaction at merchant server  **Note:** Possible values of status is 0 or 1 only  0 :Transaction received but not processed successfully at merchant server  1 :Transaction Processed successfully at merchant server. |



**Shopping Cart Sample Flow – Figure -4.0**

|  |  |
| --- | --- |
| Format :product1\_amount1\_commission1~ product2\_amount2\_commission2~ product3\_amount3\_commission3~  product1 : product code. (Provided by TPSL)  amount1 :amount of the product  commission1 :commission amount of the product | Shopping Cart Merchant |

# List of Parameters

