



The Tanzania Instant Payments System (TIPS)

API Integration Guide

June 2024

Version: 1.4

Document Control

Version	Release Date	Summary of Changes
Version 0.1	18 th Nov 2020	Draft prepared and compiled by TIPS team
Version 0.2	26 th Nov 2020	Draft finalized and released
Version 1.0	27 th Jan 2021	Updated API document
Version 1.1	19 th Jun 2021	Updated API document
Version 1.2	12 th Oct 2021	Inclusion of transfers reversal API; Enhancement of transfers flow to handle ACKNOWLEDGED transfers; Additional reason codes for payee FSP when ABORTING transfers; Change the data type of numbers from numbers to "strings"; and Addition of netTransactions field to the settlement notification.
Version 1.3	30 th Sep 2022	Inclusion of reconciliation files documentation; Addition of settlement window openingDate and closingDate fields to settlement notification; Addition of settlementWindowId field in the final notifications of transfer and transfer reversal; addition of information in transfer reversal messaging enquiry API; and Addition of reason code (88) to be used by payee FSP when ABORTING a non-existing transfer.
Version 1.4	21 st Jun 2024	• Removal of the asynchronous APIs for account lookup, directory registration, update and retrieval • Introduction of additional identifierType (WALLET and CONTROLNO) • Introduction of new use cases to support merchant payments, government payments and international money remittances • Introduction of optional additional information in the lookup and transfer APIs to provide more flexibility and scalability in integrations • Introduction of the Request to pay API • Introduction of error callback API for the transfer reversal • Introduction of new header in the transfer API • Documentation for state change for transfers, reversals and request to pay processes • Documentation of the operations guidelines • Documentation for supported use cases Note: Changes in this version are highlighted in grey and are compatible with version 1.3

Consultations

All comments and queries regarding this document should be forwarded to TIPS SUPPORT.

Table of Contents

1.	Introduction	6
1.1	Overview	6
1.2	Architectural Style	6
1.3	Aim and Intended Audience	6
2.	Processes and Flows	7
2.1	Account Lookup Service	7
2.1.1	Lookup process	7
2.1.2	Directory registration process	7
2.1.3	Directory updating process	8
2.1.4	Directory enquiry process	8
2.2	Transfer Service	9
2.2.1	Transfer process	9
2.2.2	Transfer enquiry process	10
2.2.3	Transfer state transition	10
2.3	Transfer Reversal Service	11
2.3.1	Transfer reversal process	11
2.3.2	Reversal messaging enquiry process	12
2.3.3	Reversal transfer enquiry process	13
2.3.4	Reversal state transition	13
2.4	Request to Pay Service	14
2.4.1	Request to pay process	14
2.4.2	Request to pay enquiry process	15
2.4.3	Request to pay state transition	16
2.5	Settlement Service	17
2.5.1	Settlement notification process	17
2.5.2	Reconciliation files	17
2.6	Fraud Service	17
2.6.1	Fraud registration process	17
2.6.2	Fraud updating process	18
2.6.3	Fraud enquiry process	18
2.7	Communication Messaging Service	19
2.7.1	Communication messaging process	19

3.	HTTPS Details	20
3.1	Methods	20
3.2	Request Headers	20
3.3	Response Codes.....	21
4.	API Services	23
4.1	Lookup API	23
4.1.1	GET /parties/{identifierType}/{identifier}/sync	23
4.1.2	POST /participants/{identifierType}/sync	25
4.1.3	PUT /participants/{identifierType}/{identifier}/sync	26
4.1.4	GET /participants/{identifierType}/sync	27
4.2	Transfer API.....	28
4.2.1	POST /transfers	28
4.2.2	PUT /transfers/{payerRef} (for confirmation)	31
4.2.3	GET /transfers/{payerRef} (for triggering reminders)	33
4.2.4	PUT /transfers/{payerRef} (for notification)	34
4.2.5	PUT /transfers/{payerRef}/error.....	35
4.2.6	GET /transfers/{payerRef}	36
4.3	Transfer Reversal API.....	40
4.3.1	POST /messageTransfersReversal	40
4.3.2	PUT /messageTransfersReversal/{payerRef} (for update & notification)	41
4.3.3	GET /messageTransfersReversal/{payerRef}	42
4.3.4	POST /transfersReversal	43
4.3.5	PUT /transfersReversal/{payerRef}.....	44
4.3.6	PUT /transfersReversal/{payerRef}/error.....	45
4.3.7	GET /transfersReversal/{payerRef}.....	46
4.4	Request to Pay API	48
4.4.1	POST /messageRequestPay.....	48
4.4.2	PUT /messageRequestPay/{requestPayRef}	50
4.4.3	GET /messageRequestPay/{requestPayRef}.....	52
4.5	Settlement API.....	54
4.5.1	POST /settlements.....	54
4.5.2	Reconciliation file – COMPLETE.....	56
4.5.3	Reconciliation file – INCOMPLETE	57
4.6	Fraud API	58

4.6.1	POST /fraud.....	58
4.6.2	PUT /fraud/{id}	60
4.6.3	GET /fraud.....	61
4.7	Communication Messaging API	63
4.7.1	POST /message	63
4.8	API Implementation summary	65
5.	Error Handling.....	67
6.	Appendix 1: Operations Guideline	70
	Appendix 1.1: Guide to perform lookup-out.....	70
	Appendix 1.2: Guide to perform lookup-in	71
	Appendix 1.3: Guide to perform transfer-out.....	71
	Appendix 1.4: Guide to perform transfer-in	73
	Appendix 1.5: Guide to perform transfer enquiry.....	77
	Appendix 1.6: Guide to perform reversal-out.....	78
	Appendix 1.7: Guide to perform reversal-in	81
	Appendix 1.8: Guide to perform reversal messaging enquiry.....	84
	Appendix 1.9: Guide to perform reversal transfer enquiry	86
	Appendix 1.10: Guide to perform requestPay-out.....	87
	Appendix 1.11: Guide to perform requestPay-in.....	89
	Appendix 1.12: Guide to perform request to pay enquiry.....	91
	Appendix 1.13: Guide to perform Funds-In	92
	Appendix 1.14: Guide to perform Funds-Out	94
	Appendix 1.15: Guide to perform Reconciliation.....	96
7.	Appendix 2: Supported Use Cases	98
	Appendix 2.1: Merchant Payments.....	98
	Appendix 2.2: Government payments	103
	Appendix 2.3: International Money Remittances.....	111

1. Introduction

1.1 Overview

Tanzania Instant Payments System (TIPS) is an interoperable digital payment platform operated by the Bank of Tanzania. The platform facilitates real-time interoperable funds transfers and payments across Financial Service Providers (FSPs), both banks and non-banks.

TIPS platform exposes a set of APIs and provides a web portal to facilitate interoperability among FSPs. This allows FSPs to perform various functionalities such as:

- ✓ Account (name) lookup – A service that allows the payer FSP to request payee FSP for payee account information (including name and account type) before sending a transfer.
- ✓ Transfer – A service that enables a payer FSP to transfer funds or make payment to payee FSP.
- ✓ Transfer reversal – A service that enables the payer FSP to request for reversal of a transfer and subsequently the payee FSP reversing back a transfer.
- ✓ Settlement – A facility for FSPs to settle and reconcile their financial obligations for TIPS transfers and payments.
- ✓ Messaging – A facility for FSPs to exchange non-transactional messages for various purposes including communications of requests for reversal of transfers, communications of requests for payments and others.
- ✓ Fraud management – A facility that enables FSPs to manage fraud cases in a central fraud register and prevent fraudsters from transacting on the platform.
- ✓ Others – Includes enquiring and visualizing transactions, reports and dashboards; manage TIPS settlement account and various configurations.

1.2 Architectural Style

The TIPS API implements synchronous and/or asynchronous REST API on different services depending on the nature of the process as described in the sections that follow.

The transport protocol for all the requests in TIPS API is JSON over HTTPS.

1.3 Aim and Intended Audience

The aim of this document is to provide the Financial Service Providers (FSPs) with the technical specifications as well as integration and operation guidelines of the TIPS platform.

2. Processes and Flows

This section illustrates the end-to-end processes and flows of the services offered by TIPS.

2.1 Account Lookup Service

2.1.1 Lookup process

The lookup (name lookup) process enables the payer FSP to request the payee FSP for the payee account information (including name and account type), subsequently, relaying this information to the payer customer for confirmation before sending a transfer or making payment.

The complete cycle for lookup process as shown in figure 1 below is split into two sub processes: 1) lookup-out and 2) lookup-in as well as its operations guideline documented in Appendix 1.1 and 1.2 respectively.

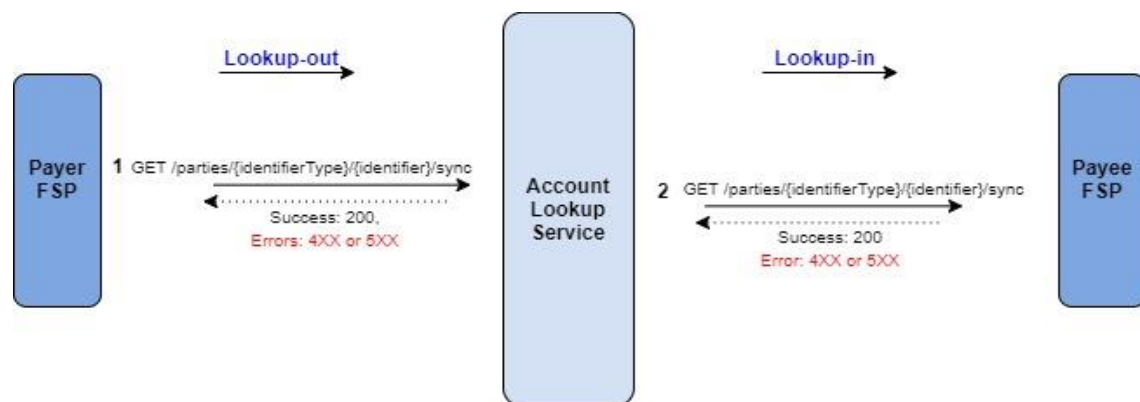


Figure 1: Lookup process – Synchronous

2.1.2 Directory registration process

TIPS platform centralizes the registration of TIPS identifiers (also known as TIPS Alias, this is a special type of identifier used by the customers to receive funds through TIPS). Therefore, the directory registration service offers central registration of the TIPS identifiers for all FSPs.

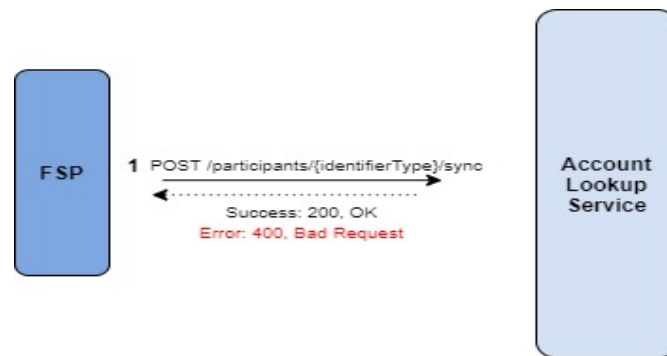


Figure 2: Directory registration process – Synchronous

2.1.3 Directory updating process

The directory updating process allows FSPs to update status of their customers' TIPS identifiers (TIPS Alias) that they have registered in TIPS central directory.



Figure 3: Directory updating process - Synchronous

2.1.4 Directory enquiry process

The directory enquiry process allows FSPs to enquire information of their customers' TIPS identifiers (TIPS Alias) that they have registered in TIPS central directory.



Figure 4: Directory enquiry process - Synchronous

2.2 Transfer Service

2.2.1 Transfer process

The transfer process enables the payer FSP to transfer funds or make payments to the payee FSP and subsequently effecting a movement of funds from the payer customer to the payee customer respectively.

The transfer process implements a two-phase commit mode (i.e. reserve and commit) with the payee FSP being the controlling node. This is to say, with reference to figure 5, the payee FSP commits first (immediately upon receipt of transfer request at step 2) and subsequently notifies other nodes to commit, which are, TIPS (upon receipt of confirmation at step 3) followed by payer FSP (upon receipt of notification at step 4).

In scenarios when no callback received by TIPS (step 3) at an agreed SLA time, TIPS sends reminders (through transaction enquiry step 2.b) to payee FSP to finalize the transfer. It is the payee FSP's responsibility to send a callback (step 3) to complete the transfer process.

The complete cycle of a transfer process as shown in figure 5 below is split in two sub processes: 1) transfer-out and 2) transfer-in as well as its operations guideline documented in Appendix 1.3 and 1.4 respectively.

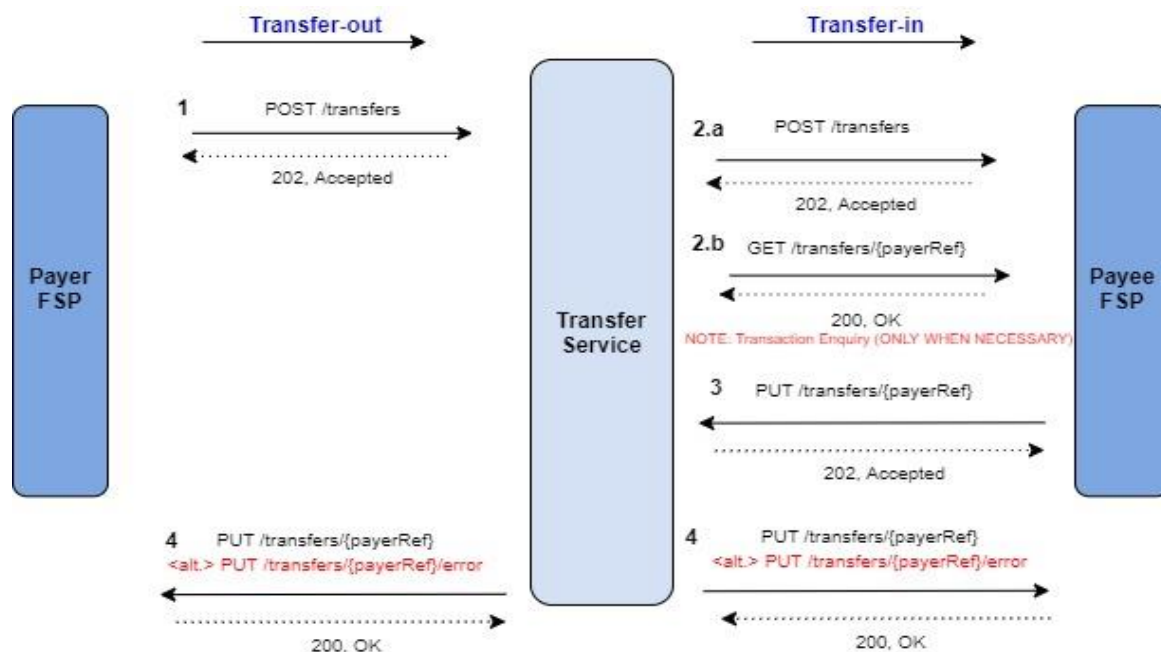


Figure 5: Transfer process

2.2.2 Transfer enquiry process

The transfer enquiry process enables the FSPs to enquire the details of the transfer in TIPS, provided, the requesting FSP is involved in that transfer as either payer or payee FSP otherwise the request will not be served. TIPS API supports transfer enquiry as shown in figure 6 below as well as its operations guideline documented in Appendix 1.5.

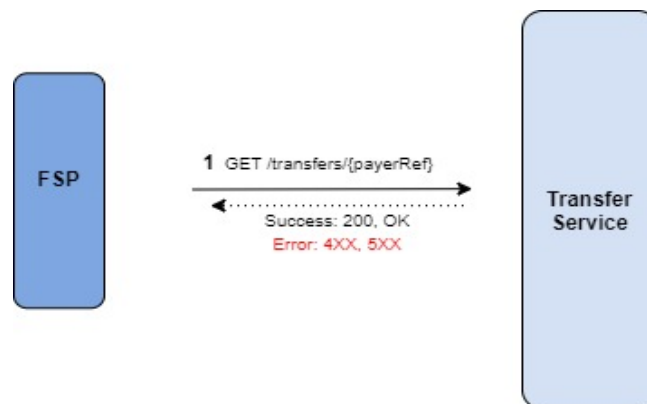


Figure 6: Transaction enquiry process

2.2.3 Transfer state transition

Transfers pass through different states during processing as shown in figure 7 below and the description for each state in the table that follows.

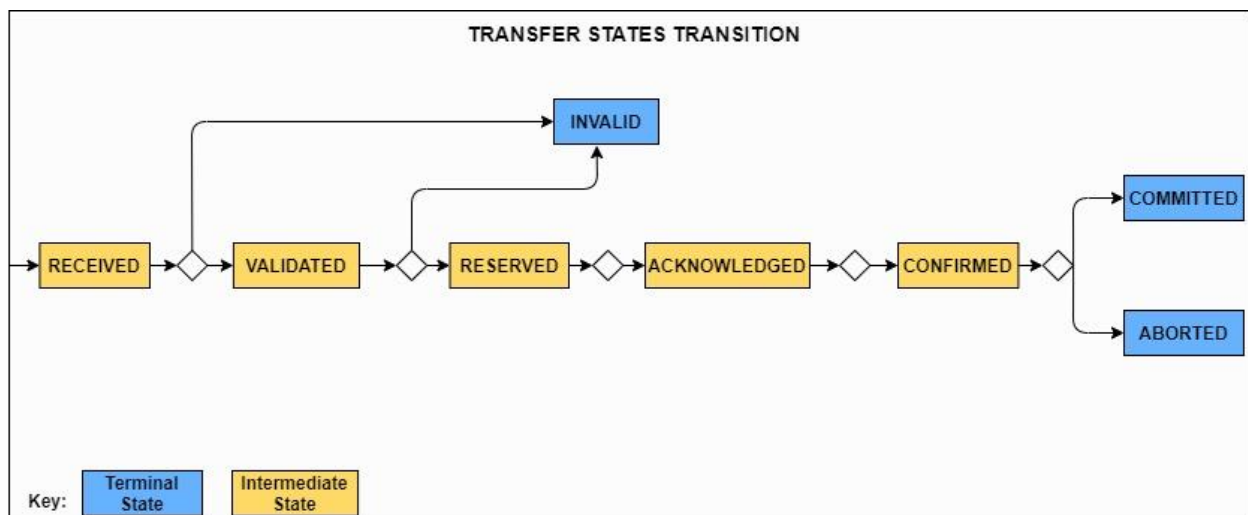


Figure 7: Transfer State Transition

The meaning of different states of the transfer in TIPS is explained in the table below;

State	Stage	Interpretation
RECEIVED	Intermediate	The transfer request from payer FSP has passed preliminary schema validations and is received by TIPS for further processing.
VALIDATED	Intermediate	The transfer request has passed all applicable schema and rules validations.
RESERVED	Intermediate	The transfer has successfully reserved funds (from payer FSPs settlement account) that gives assurance to TIPS to settle the transfer at the time of commit.
ACKNOWLEDGED	Intermediate	The payee FSP acknowledges to receive the transfer request.
CONFIRMED	Intermediate	TIPS confirms to receive a response from payee FSP with a valid instruction to commit or abort.
COMMITTED	Final/Terminal	TIPS has successfully completed and settled the transfer. The payer FSP settlement account is debited the payee FSP settlement account is credited with respect to that transfer.
INVALID	Final/Terminal	The transfer request from payer FSP has failed validations and cannot be processed further by TIPS.
ABORTED	Final/Terminal	The transfer is aborted after receiving an instruction to abort from the payee FSP.

2.3 Transfer Reversal Service

2.3.1 Transfer reversal process

The transfer reversal process enables the payer FSPs to request reversal of a particular COMMITTED transfer that subsequently results to the payee FSP returning the transfer to the payer FSP.

NOTE: Only certain categories (use cases) of transfers and payments are eligible for reversal. Details of this is covered in TIPS operating rules.

The complete cycle of a transfer reversal process as shown in figure 8 below is split into two sub processes: 1) reversal-out and 2) reversal-in, with each sub process consisting of two steps: 1) reversal messaging and 2) reversal transfer. The transfer reversal operations guideline is documented in Appendix 1.6 and 1.7.

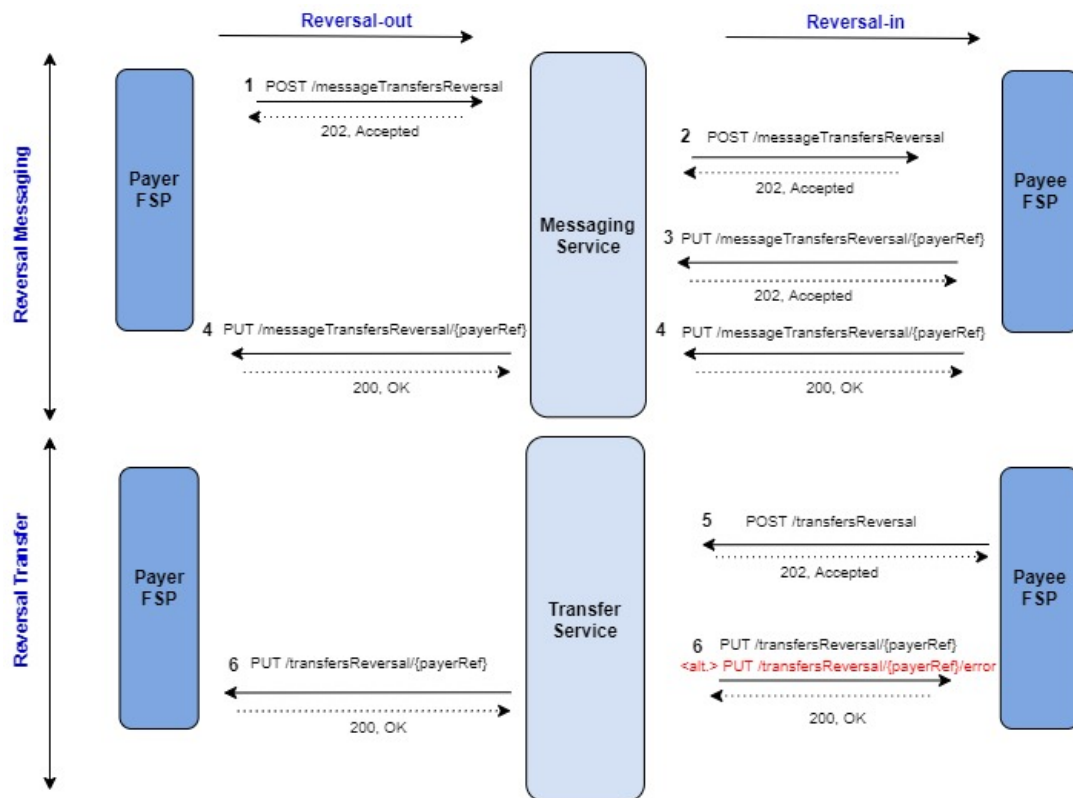


Figure 8: Transfer reversal process

2.3.2 Reversal messaging enquiry process

Reversal messaging enquiry enables the FSPs to enquire information of the reversal requests provided the requesting FSPs participated in the original transfer. TIPS API supports reversal message enquiry as shown in figure 9 below as well as its operations guideline documented in Appendix 1.8.



Figure 9: Reversal messaging enquiry process

2.3.3 Reversal transfer enquiry process

Reversal transfer enquiry enables the FSPs to enquire information of the transfer reversal provided the requesting FSPs participated in the original transfer request. TIPS API supports transfer reversal enquiry process as shown in figure 10 below as well as its operations guideline documented in Appendix 1.9.



Figure 10: Transfer reversal enquiry process

2.3.4 Reversal state transition

Transfer reversal passes through different states during processing as shown in figure 11 below and the description for each state in the table that follows.

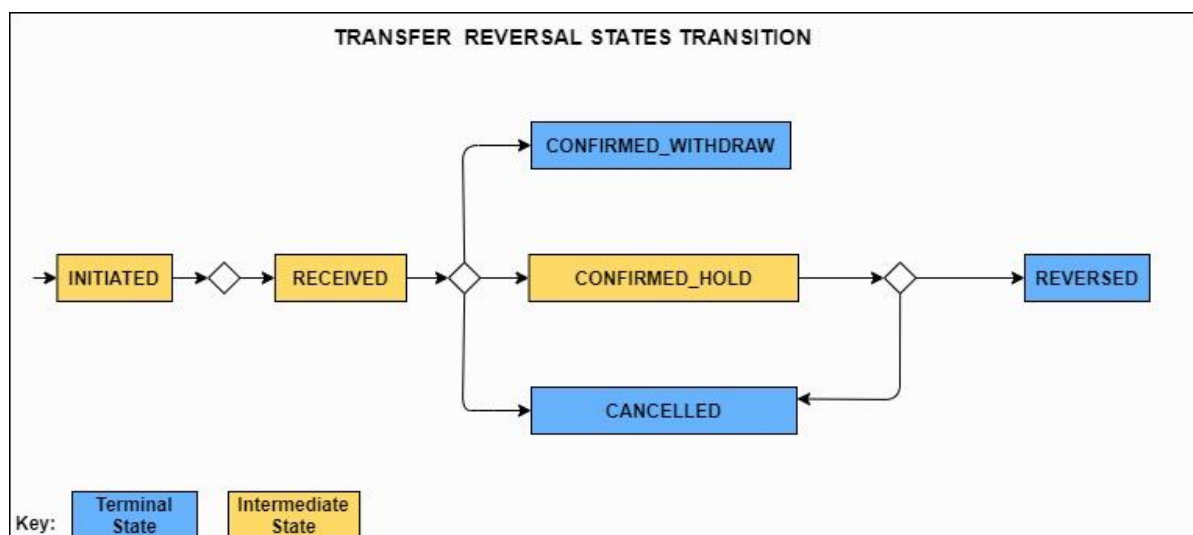


Figure 11: Transfer Reversal State Transition

The meaning of different states of the transfer reversal in TIPS is explained in the table below;

State	Stage	Details
INITIATED	Intermediate	The reversal request is successfully initiated by the payer FSP.
RECEIVED	Intermediate	The payee FSP has acknowledged receipt of the reversal request.
CONFIRMED_HOLD	Intermediate	The payee FSP has successfully hold funds from the payee customer to cover for reversal.
CONFIRMED_WITHDRAW	Final/Terminal	The payee FSP confirms funds have been withdrawn from customer account.
CANCELLED	Final/Terminal	The reversal request is cancelled by the payee FSP.
REVERSED	Final/Terminal	The reversal request is successful and complete; funds have been deducted from the payee FSP and returned to the payer FSP.

2.4 Request to Pay Service

2.4.1 Request to pay process

The request to pay is a service which enables the payee to request the payer to make a payment or a transfer. Thereafter, the payer makes a payment or transfer to the payee with respect to the request to pay.

The complete cycle of a transfer or payment with request to pay process is split in two sub processes: 1) requestPay-out and 2) requestPay-in with each sub process consisting of two steps: 1) request to pay messaging and 2) payment for request to pay. The operations guideline is documented in Appendix 1.10 and 1.11.

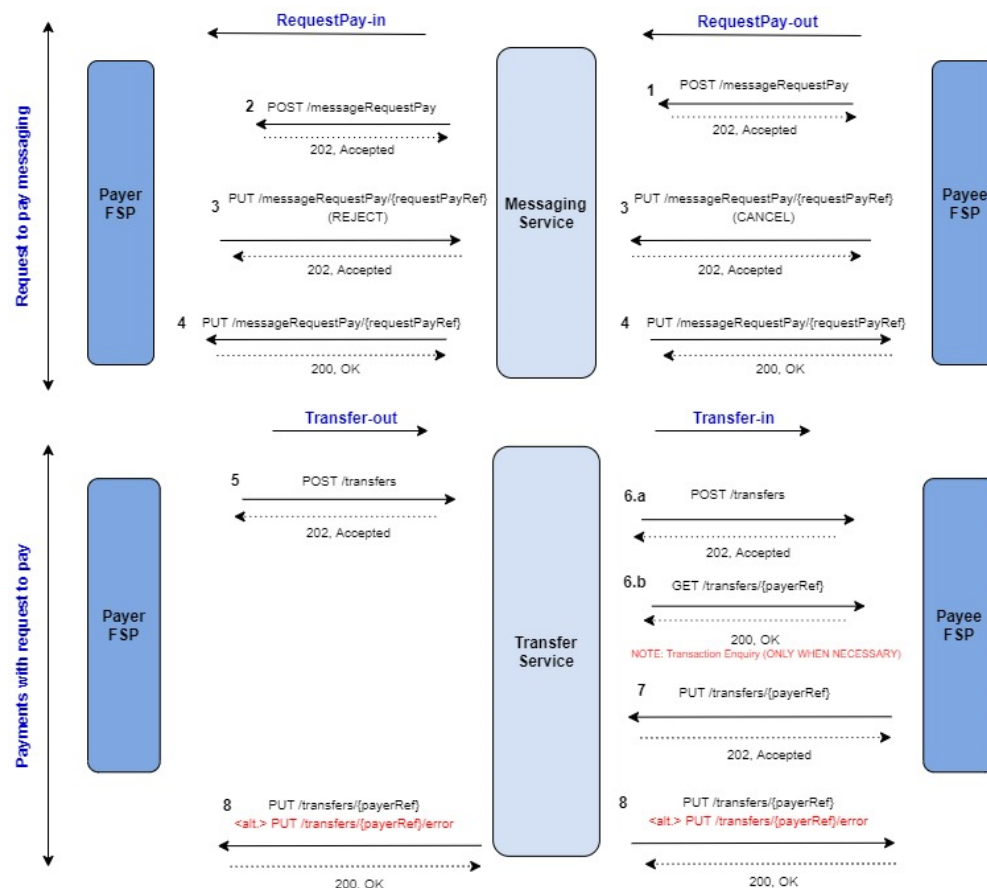


Figure 12: Transfer or Payment with request to pay

2.4.2 Request to pay enquiry process

Request to pay enquiry enables the FSPs to enquire information of the request to pay.

TIPS API supports request to pay enquiry process as shown in figure 13 below as well as its operations guideline documented in Appendix 1.12.

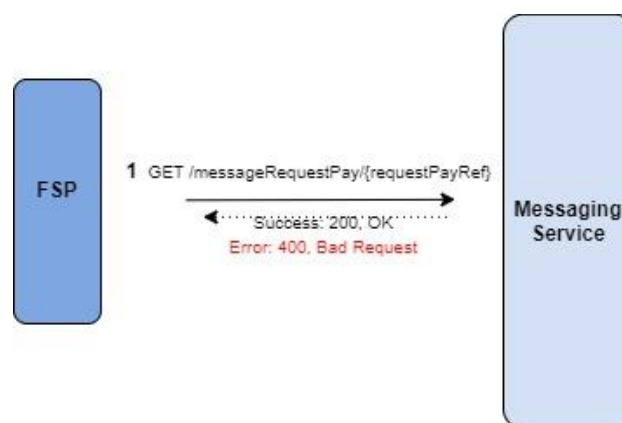


Figure 13: Request to pay enquiry

2.4.3 Request to pay state transition

The request to pay passes through different states as shown in figure 14 below and the description for each state in the table that follows:

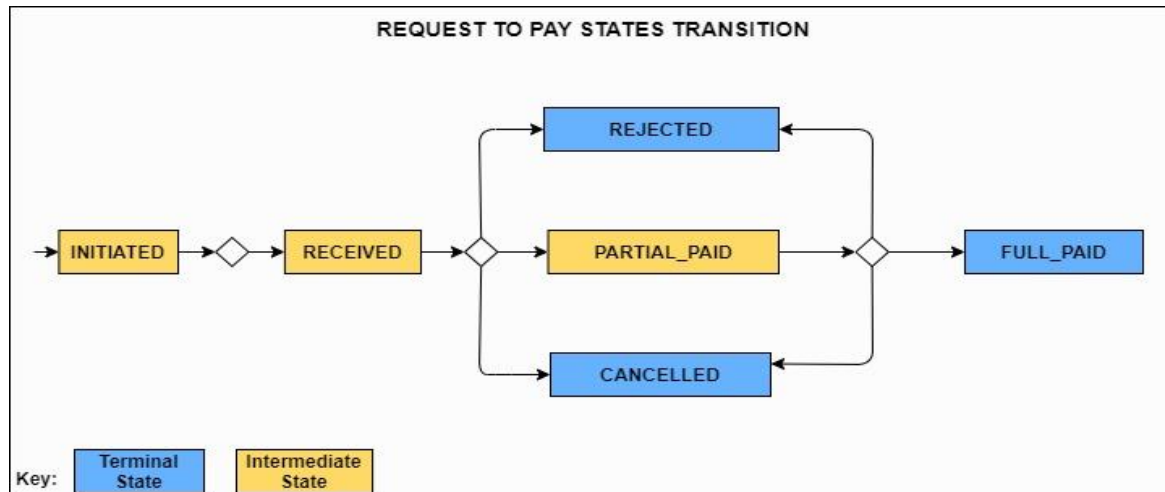


Figure 14: Request to Pay State Transition

The meaning of different states of the request to pay in TIPS is explained in the table below;

State	Stage	Details
INITIATED	Intermediate	The request to pay is successfully initiated by the payee FSP.
RECEIVED	Intermediate	The intended payer FSP has successfully acknowledged receipt of the request to pay.
PARTIAL_PAID	Intermediate	The request to pay is partially paid. More payments (or transfers) can be made with respect to that request to pay.
FULL_PAID	Final/Terminal	The request to pay is fully paid. No more transfers or payments can be made to that request to pay.
EXPIRED	Final/Terminal	The request to pay is expired. No transfers or payments are accepted.
CANCELLED	Final/Terminal	The request to pay is cancelled by the payee FSP (initiator of request to pay).
REJECTED	Final/Terminal	The request to pay is rejected by the payer FSP (original intended payer FSP of request to pay).

2.5 Settlement Service

2.5.1 Settlement notification process

The settlements notification process enables TIPS to notify the FSPs of their operations and settlement account positions in TIPS as at the end of every clearing window. The notification summarizes for each FSP, the volume and value of transactions, the net effect of transactions as well as closing position of the accounts for reconciliation purposes.

TIPS API supports settlement notification process as shown in figure 15.

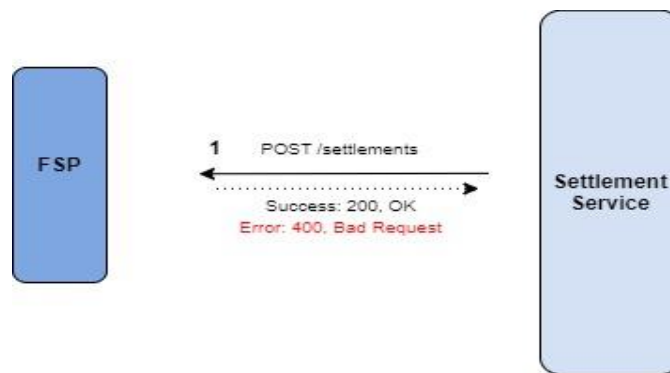


Figure 15: Settlement notification process

2.5.2 Reconciliation files

The reconciliation files provide information of each FSP transactions done in a particular clearing window. The information is provided in two files (in CSV format) for each FSP currency registered; one file showing all completed transactions (those that have affected the FSP's TIPS settlement account) and another file showing incomplete and pending transactions. The files are made available at the end of each clearing window for FSPs to download from TIPS FTP server. The guide to reconciliation is documented in Appendix 1.15

2.6 Fraud Service

2.6.1 Fraud registration process

The fraud registration service enables FSPs to register confirmed fraud cases in the central fraud register in TIPS. TIPS platform uses this register to screen and block fraudulent parties from transacting in TIPS platform. Transfers with fraudulent parties will be invalidated in TIPS.

TIPS API supports fraud registration process as shown in figure 16 below.

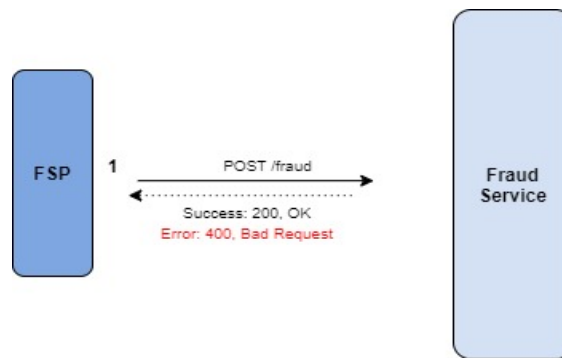


Figure 16: Fraud registration process

2.6.2 Fraud updating process

The fraud update process enables the FSPs to update the fraud central register for the fraud cases they have registered.

TIPS API supports fraud updating process as shown in figure 17.

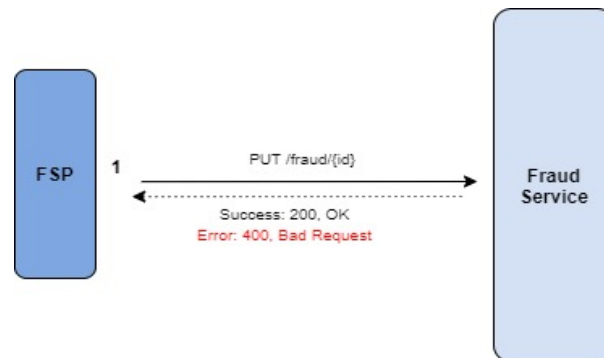


Figure 17: Fraud update process

2.6.3 Fraud enquiry process

The fraud enquiry process enables FSPs to enquire the central fraud register for the fraud cases they have registered.

TIPS API supports fraud registry enquiry process as shown in figure 18.

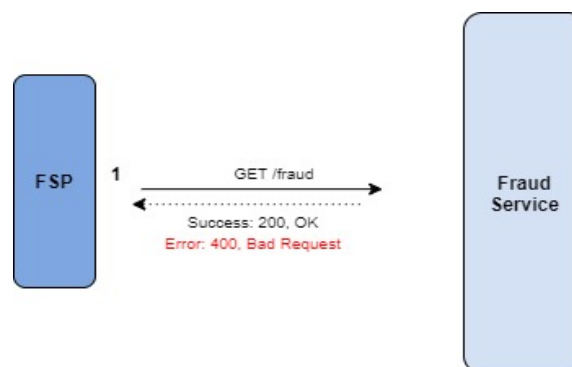


Figure 18: Fraud enquiry process

2.7 Communication Messaging Service

2.7.1 Communication messaging process

The communication messaging process enables communication or exchange of non-transactional messages (normal communication) between FSP to FSP or FSP to TIPS or TIPS to FSP(s).

TIPS API supports communication messaging process as shown in figure 19.

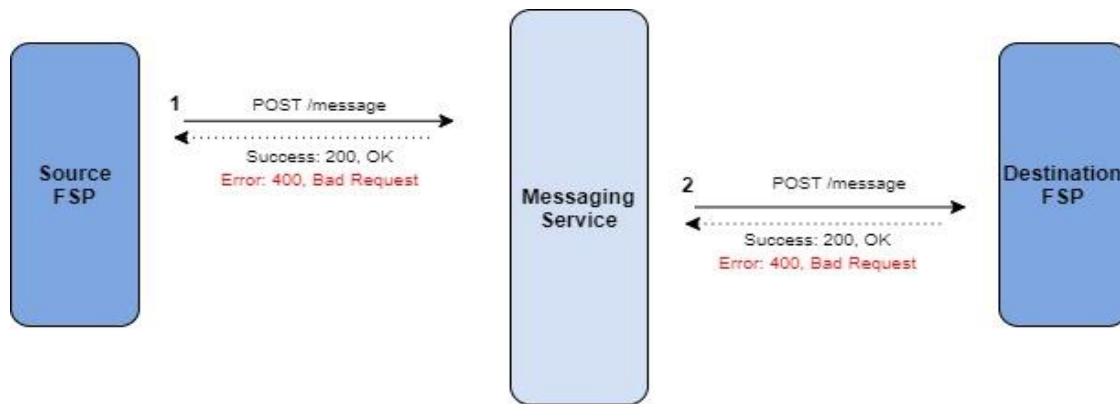


Figure 19: Communication messaging process

3. HTTPS Details

This section provides information regarding the use of the application-level protocol HTTPS.

3.1 Methods

The TIPS API implements the following methods:

- ✓ POST – To request creation of an object.
- ✓ GET – To request information about a created object.
- ✓ PUT – To update information of a created object.

NOTE: For API services that are asynchronous, the response to the POST, GET and PUT methods will come as a callback using the PUT method.

3.2 Request Headers

The API implements the following headers in the requests.

Field	Type	Example	Description
content-type	string	application/json	Indicates the specific version of the API used to send the payload body. Mandatory field.
accept	string	application/json	Indicates the version of the API the client would like the server to use. Optional field.
date	string	2019-09-11 17:25:28.67	Indicates when the request was sent. Mandatory field.
fsp-source	string	333	Indicates the source or sender of the request. Mandatory field.
fsp-destination	string	111	Indicates the destination of the request. Optional depending on nature of request.
lookup-Id	uuid	ec5bde68-f205-4619-98f8-98a281acc862	An identifier set by TIPS and returned in lookup process. Used to correlate the lookup and transfer request. Optional depending on nature of request.

request-Id	string	1201452	An identifier set by the sender of the request. Used by the fsp-source to correlate requests. Optional field.
signature	string		The signature of the request as set by the source. Used to verify the request at the destination. Optional depending on nature of the request.
Channel	String	IB	The channel used by the customer to send money or make payment. Possible channels: ✓ IB – Internet Banking ✓ MA – Mobile Application ✓ USSD - Unstructured Supplementary Service Data ✓ POS – Point of Sale Terminal at agent ✓ OTC – Over the Counter at a bank branch ✓ FT – Funds Transfer ✓ ATM – Automated Teller Machine ✓ MTO – Mobile Transfer Operator ✓ HUB – Payment Switch

3.3 Response Codes

The TIPS API implements the following HTTPS response codes:

Code	Reason	Description
SUCCESS		
200	OK	The response for a successful request. Used to mark the completion of a request.
201	Created	The request has been completed and has resulted in creation of a resource.

202	Accepted	The request has been accepted for future processing. Used to acknowledge receipt of an asynchronous request.
ERRORS		
400	Bad Request	The application cannot process the request; for example, due to malformed syntax or adding duplicates to database and other processing errors.
404	Not Found	The resource specified or information requested was not found.
500	Internal Server Error	The server has encountered error and unable to fulfil the request at that time. The client should retry.

4. API Services

This section provides detailed descriptions and examples for the requests and responses of API calls in TIPS.

4.1 Lookup API

4.1.1 GET /parties/{identifierType}/{identifier}/sync

- ✓ **Description:** This request is used to request payee customer information (name lookup)
- ✓ **Implementation:** Implemented by both TIPS and FSPs and consumed by TIPS and FSPs.
- ✓ **Request:**

```
GET /parties/MSISDN/255621804189/sync

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "004"
```

✓ Success Response:

```
HTTP/1.1 200 OK
{
  "identifierType": "MSISDN",
  "identifier": "255621804189",
  "fspId": "004",
  "fullName": " Alex Junior",
  "accountCategory": "PERSON",
  "accountType": "WALLET",
  "identity": {
    "type": "NIN",
    "value": "19501007111010000126"
  }
  "additionalInfo": {}
}
```

✓ Error Response:

```
HTTP/1.1 400 Bad Request
{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ Description of variables:

Variable	Description and Examples
URL	/parties/{identifierType}/{identifier}/sync
Headers	Described in request headers section 3.2
identifierType	This is the type of the identifier. Allowed enumerations are: • BUSINESS – identifies business and merchant numbers

	• ALIAS – identifies TIPS identifiers • MSISDN – identifies mobile phone numbers • BANK – identifies bank account numbers • WALLET – identifies wallets that are not identified by MSISDNs • CONTROLNO – identifies GePG control numbers
identifier	The account identifier. Examples values for this request are: • BUSINESS – e.g. 00300125 • ALIAS – e.g. TANAPA • MSISDN – e.g. 255621804189 • BANK – e.g. 01258796325 • WALLET – e.g. 10725542585 • CONTROLNO – e.g. 998447856741 NOTE: When passing identifiers (TIPS Aliases) that contain “@” symbol, encode the “@” symbol with %40.
fspId	The ID of the FSP owning the identifier. Each FSP will have a unique 3 characters code identification.
fullName	The full name of the account owner.
accountCategory	The category of the transacting account. Possible values are PERSON (for personal accounts), BUSINESS (for business accounts), MERCHANT (for merchant accounts) and GOVERNMENT (for government accounts)
accountType	The type of the transacting account. Possible enumerations values are: WALLET, BANK
identity	The identification of the account owner. • The “type” field can be “NIN” for National ID, while the “value” is the ID number itself without hyphens. • The “type” field can be “TIN” for Tax Id Numbers, while the “value” is the ID number itself without hyphens. NOTE: This is optional parameter
additionalInfo	This additional information that are specific for specified use cases. This applies for specific integrations ONLY. It is an object containing up to a maximum of ten key & value pairs. <pre>"additionalInfo": { "key1": "value1", ..., "key10": "value10"</pre>

	<pre>}</pre> NOTE: This is optional parameter. The actual key names and what values to pass will be available for specific use cases.
--	---

4.1.2 POST /participants/{identifierType}/sync

- ✓ **Description:** This request is used to request registration of TIPS identifiers to the TIPS directory.
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs
- ✓ **Request:**

```
POST /participants/ALIAS/sync HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "TIPS"
Request-Id: "7895435"

{
  "identifier": "MIKUMI"
}
```

- ✓ **Success Response:**

```
HTTP/1.1 200 OK

{
  "identifier": "MIKUMI",
  "status": "Active"
}
```

- ✓ **Error Response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

- ✓ **Description of variables:**

Variable	Description and Examples
URL	/participants/{identifierType}/sync
Headers	Described in request headers section 3.2
identifierType	The type of the identifier. Possible values for this request are: • ALIAS – use this to register TIPS identifiers
identifier	The account identifier. For this request:

	Set an alphanumeric string when registering the TIPS identifier (e.g. TANAPA)
Status	This field indicates the status of the identifier that is registered. It will be either active or inactive.
errorCode	The code of the error described in section 5
errorDescription	The textual description of the error described in section 5

4.1.3 PUT /participants/{identifierType}/{identifier}/sync

✓ **Description:** This request is used to update status about TIPS identifiers in the central TIPS directory.

✓ **Implementation:** Implemented by TIPS and consumed by FSPs

✓ **Request:**

```
PUT /participants/ALIAS/MIKUMI/sync
HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "TIPS"
Request-Id: "7895436"

{
  "status": "Inactive"
}
```

✓ **Success Response:**

```
HTTP/1.1 200 OK

{
  "identifier": "MIKUMI",
  "status": "Inactive"
}
```

✓ **Error Response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/participants/{identifierType}/{identifier}/sync
Headers	Described in request headers section 3.2

identifierType	The type of the identifier. Possible values for this request are: • ALIAS – use this to register TIPS identifiers
identifier	The account identifier. For this request: • Set an alphanumeric string when registering the TIPS identifier (e.g. TANAPA) NOTE: When passing identifiers (TIPS Aliases) that contain “@” symbol, encode the “@” symbol with %40.
errorCode	The code of the error described in section 5
errorDescription	The textual description of the error described in section 5

4.1.4 GET /participants/{identifierType}/sync

- ✓ **Description:** This request is used to retrieve information about TIPS identifiers registered in TIPS directory.
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs
- ✓ **Request:**

```
GET /participants/ALIAS HTTP/1.1

Accept: application/json
Content-Type: application/ json
Fsp-source: "003"
Fsp-destination:
Date:2020-10-14 10:45:50
```

- ✓ **Query parameters:**

The optional query parameters: identifier

- ✓ **Success Response:**

```
HTTP/1.1 200 OK

[
  {
    "identifier": "MIKUMI",
    "status": "Active"
  }
]
```

- ✓ **Error Response:**

```
HTTP/1.1 400 Bad Request
{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/participants/{identifierType}/sync
Headers	Described in request headers section 3.2
identifierType	The type of the identifier. Possible values for this request are: • ALIAS – use this to register TIPS identifiers
identifier	The account identifier. For this request: • Set an alphanumeric string when registering the TIPS identifier (e.g. TANAPA)
Status	This field indicates the status of the identifier that is registered. It will be either active or inactive.

4.2 Transfer API**4.2.1 POST /transfers**

✓ **Description:** This is used to requests transfer of funds or make payments in TIPS from the payer FSP to payee FSP.

✓ **Implementation:** Implemented by both TIPS and FSPs and consumed by both TIPS and FSPs

NOTE: When the transfer request comes as incoming transfer request to payee FSP, the “endUserFee” field is not included.

✓ **Request:**

```
POST /transfers HTTP/1.1

Accept: application/json
Channel: "IB"
Content-Type: application/json
Date:2020-10-14 10:46:05
Fsp-source: "003"
Fsp-destination: "004"
Lookup-Id: 9dcb27b1-90b1-4547-b7fb-beac477bf2ab

{
  "payerRef": "003-7k084ehse56",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "003",
    "fullName": "Ally Mtoni",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    },
    "additionalInfo": {}
  },
  "payee": {
```

```

    "identifierType": "MSISDN",
    "identifier": "255621804189",
    "fspId": "004",
    "fullName": " Alex Junior",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19501007111010000126"
    },
    "additionalInfo": {}
  },
  "amount": {
    "amount": "100",
    "currency": "TZS"
  },
  "endUserFee": {
    "amount": "5",
    "currency": "TZS"
  },
  "transactionType": {
    "scenario": "TRANSFER",
    "initiator": "PAYER",
    "initiatorType": "CONSUMER"
  },
  "description": "Test description",
  "additionalInfo": {}
}

```

✓ Response:

```

HTTP/1.1 202 Accepted

{
  "payerRef": "003-7k084ehse56",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

```

✓ Description of variables:

Variable	Description
URL	/transfers
Headers	Described in request headers section 3.2
payerRef	This is the transaction reference. The payer FSP sets it. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-7k084ehse56
payer	This is the object identifying the payer customer
payee	This is the object identifying the payee customer
identifierType	This is the type of the identifier. Allowed enumerations are: - BUSINESS – identifies business and merchant numbers - ALIAS – identifies TIPS identifiers

	• MSISDN – identifies mobile phone numbers • BANK – identifies bank account numbers • WALLET – identifies wallets that are not identified by MSISDNs • CONTROLNO – identifies GePG control numbers
identifier	The account identifier. Examples values for this request are: • BUSINESS – e.g. 00300125 • ALIAS – e.g. TANAPA • MSISDN – e.g. 255621804189 • BANK – e.g. 01258796325 • WALLET – e.g. 10725542585 • CONTROLNO – e.g. 998447856741 NOTE: When passing identifiers (TIPS Aliases) that contain “@” symbol, encode the “@” symbol with %40.
fspId	The ID of the FSP owning the identifier. Each FSP will have a unique 3 characters code identification.
fullName	The full name of the account owner.
accountCategory	The category of the transacting account. Possible values are PERSON (for personal accounts), BUSINESS (for business accounts), MERCHANT (for merchant accounts) and GOVERNMENT (for government accounts)
accountType	The type of the transacting account. Possible values are WALLET, BANK
identity	The identification of the account owner. • The “type” field can be “NIN” for National ID, while the “value” is the ID number itself without hyphens. • The “type” field can be “TIN” for Tax Id Numbers, while the “value” is the ID number itself without hyphens. NOTE: This is optional parameter
amount	This is the object defining the transfer amount. It contains: • amount – the transfer amount. No decimals for wallet transfers. • currency – the currency of transfer amount (TZS).
endUserFee	These are end user fees charged to the customer by the payee FSP • amount – the fee amount. • currency – the currency of fee amount (TZS).

transactionType	This object defines the type of transaction. It contains • scenario – nature of the transaction. Possible values are: TRANSFER (for fund transfers) or PAYMENT (for payments to merchants or government), PAYMENT_REQUESTPAY (payments with request to pay), TRANSFER_IMT (international money remittances). • initiator – who initiates the transaction request. Possible values are: PAYER (push payment) or PAYEE (pull payment) • initiatorType – type of the initiator. Possible values are: CONSUMER or BUSINESS
additionalInfo	This additional information that are specific for specified use cases. This applies for specific integrations ONLY. It is an object containing up to a maximum of ten key & value pairs. <pre>"additionalInfo": { "key1": "value1", ..., "key10": "value10" }</pre> NOTE: This is optional parameter. The actual key names and what values to pass will be available for specific use cases.
description	This is the description of the transaction.
status	The status of the request at the request recipient side. To show acknowledgement
datetime	Date and time of acknowledging the request

4.2.2 PUT /transfers/{payerRef} (for confirmation)

- ✓ **Description:** This request is used by payee FSP as a callback to confirm to TIPS to either commit or abort a transfer.
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs
- ✓ **Request:**

```
PUT /transfers/003-7k084ehse56 HTTP/1.1

Accept: application/json
Content-type: application/json
Date: 2020-10-14 10:46:10
Fsp-source: "004"
Fsp-destination: "003"

{
  "payeeRef": "004-xyz123hy999",
  "transferState": "COMMITTED ",
  "reasonCode": "60",
  "additionalInfo": {}
}
```

✓ **Response:**

```
HTTP/1.1 202 Accepted

{
  "payerRef": "003-7k084ehse56",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}
```

✓ **Description of variables:**

Variable	Description
URL	/transfers/{payerRef}
Headers	Described in request headers section 3.2
payerRef	This is the payer transaction reference. The payer FSP sets it. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-7k084ehse56
payeeRef	This is the payee transaction reference. The payee FSP sets it. The format of this reference is <payee fspId>-<payee system ref.> e.g. 004-xyz123hy999
transferState	This is the state of transaction from payee. The values can be COMMITTED (to instruct switch to commit successfully) or ABORTED (to instruct switch to abort transaction)
reasonCode	This is the code to describe reason for commit or abort. For COMMITTED always use reasonCode=60 while for ABORTED use codes as described below.
additionalInfo	This additional information that are specific for specified use cases. This applies for specific integrations ONLY. It is an object containing up to a maximum of ten key & value pairs. <pre>"additionalInfo": { "key1": "value1", ..., "key10": "value10" }</pre> NOTE: This is optional parameter. The actual key names and what values to pass will be available for specific use cases.

✓ **Reason Codes:**

CODE	TRANSFER STATE	REASON DESCRIPTION
60	COMMITTED	The transaction is successfully committed
75	ABORTED	Customer KYC Issues
76	ABORTED	Fraud Related Issues
77	ABORTED	Customer Maximum Limit Exceeded

78	ABORTED	Incorrect Account
79	ABORTED	Closed Account
80	ABORTED	Blocked Account
81	ABORTED	Frozen Account
82	ABORTED	Invalid date
83	ABORTED	Forbidden Transaction
84	ABORTED	Duplicate Transfer
85	ABORTED	Incorrect Transfer Amount
86	ABORTED	Payee FSP Internal Errors
87	ABORTED	Account Currency Mismatch
88	ABORTED	Transfer not found/processed/received

NOTE: In circumstances the transfer requests fails to be delivered to payee FSP, TIPS will still want to be sure that the payee FSP has indeed not received or processed that transfer request so it will send an enquiry to payee FSP. The payee FSP will be required to confirm by sending a call back to ABORT with reason code 88 and a dummy payee reference (preferably a uuid or timestamp in milliseconds). See example below

```
{
  "payeeRef": "<fspId>-<uuid> or <fspId>-<timestamp_in_miliseconds>",
  "transferState": "ABORTED ",
  "reasonCode": "88"
}
```

4.2.3 GET /transfers/{payerRef} (for triggering reminders)

- ✓ **Description:** This request is used by TIPS to trigger reminders (through status enquiry to payee FSP) so that they send a callback to commit or abort a particular transfer.
- ✓ **Implementation:** Implemented by FSPs and consumed by TIPS
- ✓ **Request:**

```
GET /transfers/003-7k084ehse56 HTTP/1.1

Accept: application/json
Content-Type: application/json
Fsp-source: "tips"
Fsp-destination: "003"
Date:2020-10-15 10:22:12
```

- ✓ **Success Response:**

```
HTTP/1.1 200 OK
{
}
```

NOTE: The response may contain a payload (details of a transfer), but TIPS will not validate

✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "3208",
    "errorDescription": "Transfer Ref not found"
  }
}
```

4.2.4 PUT /transfers/{payerRef} (for notification)

✓ **Description:** This request is used by TIPS as a callback to notify payer and payee FSPs on a successful completion (COMMITTED) of a transfer.

✓ **Implementation:** Implemented by FSPs and consumed by TIPS

✓ **Request:**

```
PUT /transfers/003-7k084ehse56 HTTP/1.1

Accept: application/json
Content-type: application/json
Date: 2020-10-14 10:46:13
Fsp-source: "TIPS"
Fsp-destination: "003"
{
  "payeeRef": "004-xyz123hy999",
  "switchRef": "021950bd-7e98-4b67-adb9-321eb0bad792",
  "completedTimestamp": "2020-10-14 10:46:13",
  "transferState": "COMMITTED",
  "settlementWindowId": "115",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "003",
    "fullName": "Ally Mtoni",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    },
    "additionalInfo": {}
  },
  "payee": {
    "identifierType": "MSISDN",
    "identifier": "255621804189",
    "fspId": "004",
    "fullName": " Alex Junior",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19501007111010000126"
    },
    "additionalInfo": {}
  },
  "additionalInfo": {}
}
```

✓ **Response:**

```
HTTP/1.1 200 OK
{
  "payerRef": "003-7k084ehse56",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}
```

✓ **Description of request variables:**

Variable	Description
URL	/transfers/{payerRef}
Headers	Described in request headers section 3.2
payerRef	This is the transaction reference. The payer FSP sets it. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-7k084ehse56
payeeRef	This is the payee transaction reference. The payee FSP sets it. The format of this reference is <payee fspId>-<payee system ref.> e.g. 004-xyz123hy999
switchRef	The unique identification of each transaction in the switch. It is generated by the switch.
completedTimestamp	Time the switch has completed processing the transaction.
transferState	This is the state of transaction from payee. The values can be COMMITTED (to instruct switch to commit successfully)
settlementWindowId	The unique ID of the settlement window in which the transfer completed

4.2.5 PUT /transfers/{payerRef}/error

✓ **Description:** This request is used by TIPS as a callback to notify payer or payee FSPs on the errors incurred during processing of a transfer that resulted into termination of the transfer.

✓ **Implementation:** Implemented by FSPs and consumed by TIPS

✓ **Request:**

```
PUT /transfers/003-7k084ehse56/error HTTP/1.1
Accept: application/json
Content-Type: application/json
Fsp-source: "TIPS"
Fsp-destination: "003"
Date:2020-10-14 10:46:13
{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Response:**

```

HTTP/1.1 200 OK

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:46:07
Fsp-source: "tips"
Fsp-Destination: "003"

{
  "payerRef": "003-7k084ehse56",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

```

✓ **Description of variables:**

Variable	Description
URL	/transfers/{payerRef}
Headers	Described in request headers section 3.2
payerRef	This is the transaction reference. The payer FSP sets it. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-7k084ehse56
errorCode	The code of the error described in section 5
errorDescription	The textual description of the error described in section 5

4.2.6 GET /transfers/{payerRef}

- ✓ **Description:** This request is used by FSPs to request information about a transfer (TIPS transaction status enquiry).
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs
- ✓ **Request:**

```

GET /transfers/003-7k084ehse56 HTTP/1.1

Accept: application/json
Content-Type: application/ json
Fsp-source: "003"
Fsp-destination: "tips"
Date:2020-10-15 10:22:12

```

✓ **Success Response:**

```

HTTP/1.1 200 OK
{
  "payerRef": "003-7k084ehse56",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "003",
    "fullName": "Ally Mtoni",
    "accountCategory": "PERSON",

```

```

    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    },
    "additionalInfo": {}
  },
  "payee": {
    "identifierType": "MSISDN",
    "identifier": "255621804189",
    "fspId": "004",
    "fullName": " Alex Junior",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19501007111010000126"
    },
    "additionalInfo": {}
  },
  "amount": {
    "amount": "100",
    "currency": "TZS"
  },
  "endUserFee": {
    "amount": "5",
    "currency": "TZS"
  },
  "transactionType": {
    "scenario": "TRANSFER",
    "initiator": "PAYER",
    "initiatorType": "CONSUMER"
  },
  "description": "Test description"
  "lookupId": "9dcb27b1-90b1-4547-b7fb-beac477bf2ab"
  "transferState": "COMMITTED",
  "payeeRef": "004-xyz123hy999",
  "switchRef": "349e98a6-94d1-4de8-badf-64b0e8b09580",
  "transferDate": "2020-10-14 10:46:05",
  "completedDate": "2020-10-14 10:46:11",
  "transactionCategory": {
    "useCase": "PERSON-PERSON",
    "subUseCase": "WALLET-WALLET",
    "description": "P2P Wallet to Wallet"
  },
  "settlementWindow": {
    "id": "2",
    "description": "window for 14 Oct 2020"
  },
  "additionalInfo": {}
}

```

✓ Error response:

HTTP/1.1 400 Bad Request

```

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}

```

✓ **Description of variables:**

Variable	Description
URL	/transfers/{payerRef}
Headers	Described in request headers section 3.2
payerRef	This is the transaction reference. The payer FSP sets it. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-7k084ehse56
payer	This is the object identifying the payer customer
payee	This is the object identifying the payee customer
identifierType	This is the type of the identifier. Allowed enumerations are: - BUSINESS – identifies business and merchant numbers - ALIAS – identifies TIPS identifiers - MSISDN – identifies mobile phone numbers - BANK – identifies bank account numbers - WALLET – identifies wallets that are not identified by MSISDNs - CONTROLNO – identifies GePG control numbers
identifier	The account identifier. Examples values for this request are: - BUSINESS – e.g. 00300125 - ALIAS – e.g. TANAPA - MSISDN – e.g. 255621804189 - BANK – e.g. 01258796325 - WALLET – e.g. 10725542585 - CONTROLNO – e.g. 998447856741 NOTE: When passing identifiers (TIPS Aliases) that contain “@” symbol, encode the “@” symbol with %40.
fspId	The ID of the FSP owning the identifier. Each FSP will have a unique 3 characters code identification.
fullName	The full name of the account owner.
accountCategory	The category of the transacting account. Possible values are PERSON (for personal accounts), BUSINESS (for business accounts), MERCHANT (for merchant accounts) and GOVERNMENT (for government accounts)
accountType	The type of the transacting account. Possible values are WALLET, BANK

identity	The identification of the account owner. • The “type” field can be “NIN” for National ID, while the “value” is the ID number itself without hyphens. • The “type” field can be “TIN” for Tax Id Numbers, while the “value” is the ID number itself without hyphens. NOTE: This is optional parameter
amount	This is the object defining the transfer amount. It contains: • amount – the transfer amount. No decimals for wallet transfers. • currency – the currency of transfer amount (TZS).
endUserFee	These are end user fees charged to the customer by the payee FSP • amount – the fee amount. • currency – the currency of fee amount (TZS).
transactionType	This object defines the type of transaction. It contains • scenario – nature of the transaction. Possible values are: TRANSFER (for transfers) or PAYMENT (for payments to merchants and government), PAYMENT_REQUESTPAY (payments with request to pay), TRANSFER_IMT (international money remittances). • initiator – who initiates the transaction request. Possible values are: PAYER (push payment) or PAYEE (pull payment) • initiatorType – type of the initiator. Possible values are: CONSUMER or BUSINESS
description	This is the description of the transaction.
Lookup-Id	A unique identifier that correlates lookup and transfer.
transferState	The current state of a transfer in TIPS at the time of retrieving a transfer. Possible states: RECEIVED, VALIDATED, INVALID, RESERVED, CONFIRMED, ABORTED, COMMITTED
transferDate	The date the transfer received in TIPS.
completedDate	The date the transfer reaches terminal states, either INVALID, ABORTED or COMMITTED.

transactionCategory	This object defines the category of the transfer. Categorization is based on use case and sub-use cases.
settlementWindow	This object defines the settlement/clearing window the transfer was assigned in TIPS. It is used for reconciliation.
additionalInfo	This additional information that are specific for specified use cases. This applies for specific integrations ONLY. It is an object containing up to a maximum of ten key & value pairs. <pre>"additionalInfo": { "key1": "value1", ..., "key10": "value10" }</pre> NOTE: This is optional parameter. The actual key names and what values to pass will be available for specific use cases.

4.3 Transfer Reversal API

4.3.1 POST /messageTransfersReversal

- ✓ **Description:** This request is used by the payer FSP to initiate communication of transfer reversal request to payee FSP.
- ✓ **Implementation:** Implemented by both TIPS and FSPs and consumed by both TIPS and FSPs.
- ✓ **Request:**

```
POST /messageTransfersReversal HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "004"

{
  "payerReversalRef": "003-yujnjth-678",
  "payerRef": "003-7k084ehse56",
  "payeeRef": "004-xyz123hy999",
  "switchRef": "021950bd-7e98-4b67-adb9-321eb0bad792",
  "amount": {
    "amount": "100.00",
    "currency": "TZS"
  },
  "reversalReason": "Sent funds twice"
}
```

- ✓ **Success Response:**

```
HTTP/1.1 202 Accepted
```

✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/messageTransfersReversal
Headers	Described in headers section 3.2
payerReversalRef	This is the reversal reference that is set by the payer FSP who is requesting for a transfer reversal. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003- yujnhh7890
payerRef	This is the payer reference of the original transaction.
payeeRef	This is the payee reference of the original transaction.
switchRef	This is the switch reference of the original transaction.
amount	This is the object defining the original transfer amount containing amount and currency.
reversalReason	The reason for requesting a transfer reversal.

4.3.2 PUT /messageTransfersReversal/{payerRef} (for update & notification)

✓ **Description:** This request is used by 1) the payee FSP as a callback to TIPS to update the state of the transfer reversal request 2) TIPS to notify the original payer FSP of the transfer about the status of the transfer reversal.

✓ **Implementation:** Implemented by TIPS and FSPs and consumed by TIPS and FSPs.

✓ **Request:**

```
PUT /messageTransfersReversal/{payerRef} HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "004"
Fsp-destination: "003"

{
  "reversalState": "CONFIRMED_HOLD",
  "reversalReason": "successfull hold funds"
}
```

✓ **Success Response:**

```
HTTP/1.1 200 OK
```

✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/messageTransfersReversals/{payerRef}
Headers	Described in headers section 3.2
payerRef	This is the payer reference of the original transaction.
reversalState	This is the state of the transfer reversal as processed by the payee FSP. Possible update of states are: • CONFIRMED_HOLD – payee FSP confirms to hold the funds so that it cannot be utilized by payee customer • CONFIRMED_WITHDRAW – payee FSP confirms the amount already withdrawn by payee customer • CANCELLED – reversal canceled by payee FSP.
reversalReason	This is the reason for changing the state of the transfer reversal request.

4.3.3 GET /messageTransfersReversal/{payerRef}✓ **Description:** This request is used for the enquiry of a transfer reversal process.✓ **Implementation:** Implemented by TIPS and consumed by FSPs.✓ **Request:**

```
GET /messageTransfersReversal/{payerRef}
HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "TIPS"
```

✓ **Success Response:**

```
HTTP/1.1 200 OK
{
  "reversalRef": "002-m3791650s-ctc4",
  "payerRef": "002-7eab2c4a-204c-4933",
  "payeeRef": "003-f7c71870-4099-43c4-aca0",
  "switchRef": "e5fddd99-9f5c-434f-a05c",
  "amount": {
    "amount": 10,
    "currency": "TZS"
  },
  "completedTimestamp": "2022-06-14 10:44:27",
  "reversalState": "INITIATED",
  "reversalReason": "Sent by Mistake"
}
```

✓ **Error response:**

```
HTTP/1.1 400 Bad Request
{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/messageTransferReversals/{payerRef}
Headers	Described in headers section 3.2
reversalRef	The reversal reference set by the original payer of the transfer
payerRef	This is the payer reference of the original transaction.
payeeRef	This is the payee reference of the original transaction.
switchRef	This is the switch reference of the original transaction.
amount	Transfer amount
completedTimestamp	Time the reversal completed
reversalState	The current state of the transfer reversal.
reversalReason	The reason for update of a transfer reversal.

4.3.4 POST /transfersReversal

- ✓ **Description:** This request is used by the payee FSP of the original transfer to reverse a particular transfer to the original payer FSP of the transfer.
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs.

✓ **Request:**

```
POST /transfersReversal HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "004"
Fsp-destination: "003"
{
  "payerRef": "003-yujnhh7890",
  "payeeReversalRef": "004-321eb0bad792",
  "reversalState": "REVERSED",
  "reversalReason": "string"
}
```

✓ **Success Response:**

```
HTTP/1.1 202 Accepted
```

✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/transfersReversals
Headers	Described in headers section 3.2
payerRef	This is the payer reference of the original transaction.
payeeReversalRef	This is the reversal reference that is set by the payee FSP who is reversing a transfer. The format of this reference is <payee fspId>-<payee system ref.> e.g. 004- 321eb0bad792
reversalState	This is the state of the transfer reversal as processed by the payee FSP. Possible update of states are: REVERSED – payee FSP has successfully reversed the funds
reversalReason	This is the reason for changing the state of the transfer reversal request.

4.3.5 PUT /transfersReversal/{payerRef}

✓ **Description:** This request is used by TIPS to inform FSPs on the successful completion of the reversal of a transfer.

✓ **Implementation:** Implemented by FSPs and consumed by TIPS.

✓ **Request:**

```
PUT /transfersReversal/{payerRef} HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "tips"
Fsp-destination: "003"
```

```
{
  "switchRef": "021950bd-7e98-4b67-adb9-321eb0bad792",
  "payeeRef": "004-xyz123hy999",
  "switchReversalRef": "021950bd-7e98-4b67-adb9-321eb0bad792",
  "payerReversalRef": "003-yujnhh7890",
  "payeeReversalRef": "004-321eb0bad792",
  "reversalState": "REVERSED",
  "completedTimestamp": "2020-10-14 10:46:06",
  "settlementWindowId": 115
}
```

✓ Success Response:

```
HTTP/1.1 200 OK
```

✓ Error response:

```
HTTP/1.1 400 Bad Request
```

```
{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ Description of variables:

Variable	Description and Examples
URL	/transfersReversals/{payerRef}
Headers	Described in request headers section 3.2
payerRef	This is the payer reference of the original transaction.
switchRef	This is the switch reference of the original transaction.
payeeRef	This is the payee reference of the original transaction.
switchReversalRef	This is the switch reference of the reverse transaction.
payerReversalRef	This is the reversal reference that is set by the payer FSP who is requesting for a transfer reversal. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-yujnhh7890
payeeReversalRef	This is the reversal reference that is set by the payee FSP who is reversing a transfer. The format of this reference is <payee fspId>-<payee system ref.> e.g. 004-321eb0bad792
reversalState	This is the state of the transfer reversal as processed by the TIPS. Possible update of states are: REVERSED – TIPS has successfully reversed the funds
completedTimestamp	The time the reversal transfer completed in TIPS.
settlementWindowId	The unique ID of the settlement window in which the transfer completed

4.3.6 PUT /transfersReversal/{payerRef}/error

- ✓ **Description:** This request is used by TIPS to inform FSPs on the errors incurred during reversal of a transfer.
- ✓ **Implementation:** Implemented by FSPs and consumed by TIPS.

✓ **Request:**

```
PUT /transfersReversal/{payerRef}/error

Accept: application/json
Content-Type: application/json
Fsp-source: "TIPS"
Fsp-destination: "003"
Date: 2020-10-14 10:46:13
{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Response:**

```
HTTP/1.1 200 OK

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:46:07
Fsp-source: "003"
Fsp-Destination: "tips"
{
  "payerRef": "003-7k084ehse56",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}
```

✓ **Description of variables:**

Variable	Description
URL	/transfersReversals/{payerRef}/error
Headers	Described in request headers section 3.2
payerRef	This is the transaction reference. The payer FSP sets it. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-7k084ehse56
errorCode	The code of the error described in section 5
errorDescription	The textual description of the error described in section 5

4.3.7 GET /transfersReversal/{payerRef}

✓ **Description:** This request is used for the enquiry of a transfer reversal.

✓ **Implementation:** Implemented by TIPS and consumed by FSPs.

✓ **Request:**

```
GET /transfersReversal/{payerRef} HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "tips"
```

✓ **Success Response:**

```

HTTP/1.1 200 OK

{
  "switchRef": "021950bd-7e98-4b67-adb9-321eb0bad792",
  "payeeRef": "004-xyz123hy999",
  "switchReversalRef": "021950bd-7e98-4b67-adb9-321eb0bad792",
  "payerReversalRef": "003-yujnhh7890",
  "payeeReversalRef": "004-321eb0bad792",
  "reversalState": "REVERSED",
  "completedTimestamp": "string",
  "amount": {
    "amount": 10,
    "currency": "TZS"
  },
  "settlementWindowId": "115",
  "newPayerRef": "004-321eb0bad792",
  "newPayeeRef": "003-yujnhh7890"
}

```

✓ **Error response:**

```

HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}

```

✓ **Description of variables:**

Variable	Description and Examples
URL	/transfersReversals/{payerRef}
Headers	Described in headers section 3.2
payerRef	This is the payer reference of the original transaction.
switchRef	This is the switch reference of the original transaction.
payeeRef	This is the payee reference of the original transaction.
switchReversalRef	This is the switch reference of the reverse transaction.
payerReversalRef	This is the reversal reference that is set by the payer FSP who is requesting for a transfer reversal. The format of this reference is <payer fspId>-<payer system ref.> e.g. 003-yujnhh7890
payeeReversalRef	This is the reversal reference that is set by the payee FSP who is reversing a transfer. The format of this reference is <payee fspId>-<payee system ref.> e.g. 004-321eb0bad792
reversalState	This is the state of the transfer reversal as processed by the TIPS. Possible update of states are: REVERSED – TIPS has successfully reversed the funds
completedTimestamp	The time the reversal transfer completed in TIPS.

4.4 Request to Pay API

4.4.1 POST /messageRequestPay

- ✓ **Description:** This request is used by the payee FSP to initiate a request to pay to the payer FSP.
- ✓ **Implementation:** Implemented by both TIPS and FSPs and consumed by both TIPS and FSPs.
- ✓ **Request:**

```
POST /messageRequestPay HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "004"
{
  "requestPayRef": "003-oYVl3qcZdm4Z3",
  "paymentRef": "tyhkfffhjy2345",
  "paymentDescription": "request to pay your bill",
  "paymentOption": "EXACT",
  "paymentDueDate": "2020-10-14 10:46:05",
  "amount": {
    "amount": "2300",
    "currency": "TZS"
  },
  "payee": {
    "identifierType": "MSISDN",
    "identifier": "255621804189",
    "fspId": "003",
    "fullName": "MAGANGA SUPPLIES",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19501007111010000126"
    }
  },
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "004",
    "fullName": "JOHN JOHN",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    }
  }
}
```

- ✓ **Success Response:**

```
HTTP/1.1 202 Accepted

{
  "requestPayRef": "003-oYVl3qcZdm4Z3",
  "paymentRef": "tyhkfffhjy2345",
  "paymentDescription": "request to pay your bill",
  "paymentOption": "EXACT",
  "paymentDueDate": "2020-10-14 10:46:05",
  "amount": {
```

```

    "amount": "2300",
    "currency": "TZS"
  },
  "payee": {
    "identifierType": "MSISDN",
    "identifier": "255621804189",
    "fspId": "003",
    "fullName": "MAGANGA SUPPLIES",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19501007111010000126"
    }
  },
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "004",
    "fullName": "JOHN JOHN",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    }
  },
  "requestPayState": "INITIATED",
  "requestPayStateReason": "Successfully received by TIPS",
  "createdDate": "2020-10-14 10:45:46"
}

```

✓ Error response:

```

HTTP/1.1 400 Bad Request
{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}

```

✓ Description of variables:

Variable	Description and Examples
URL	/messageRequestPay
Headers	Described in headers section 3.2
requestPayState	The current state the request to pay is in. At the state of initiation, it will have state of INITIATED.
requestPayStateReason	The descriptive meaning of the state of the request to pay.
createdDate	The date the request to pay is received by TIPS.
requestPayRef	This is the request to pay reference that is set by the initiator of the request to pay (payee FSP). The format of this reference is <payee fspId>-<payee system ref.> e.g. 003-oYVl3qcZdm4Z3

paymentRef	This is the payment reference set by the payee FSP.
paymentDescription	This is the description of the request to pay.
paymentOption	This is the payment option for the request to pay. The possible values are: • EXACT – Payment must be made at once. The amount must be exact as that set in the request to pay. • FULL – Payment must be made at once. The amount must be at least that set in the request to pay. • PARTIAL – Several payments allowed. Total amount paid must not exceed that set in the request to pay. • UNLIMITED – Several payments allowed. No limitations on the total amount paid.
paymentDueDate	This is the date the request to pay expires, after that date no payments are allowed to that request to pay.
amount	This is the object defining the transfer amount. It contains: • amount – the request to pay amount. • currency – the currency (TZS).
payee	This is the object identifying the payee customer. The recipient of the payments for that request to pay.
payer	This is the object identifying the intended payer customer. The intended original payer of the request to pay. NOTE: Any payer is allowed to make payment to the request to pay provided they have the request to pay references.

4.4.2 PUT /messageRequestPay/{requestPayRef}

- ✓ **Description:** This request is used by initiator of request to pay (payee FSP) to update or cancel the request to pay. It is also used by the original intended payer FSP to reject the request to pay.
- ✓ **Implementation:** Implemented by both TIPS and FSPs and consumed by both TIPS and FSPs.
- ✓ **Request:**

```
PUT /messageRequestPay/{requestPayRef} HTTP/1.1
```

```
Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "004"
```

```
{
  "paymentDueDate": "2020-12-14 10:45:44",
```

```

"requestPayState": "CANCELLED",
"requestPayReason": "Already Paid"
}

```

Note: In this request, you cannot submit all the three parameter at once. Either submit 1) paymentDueDate alone or 2) requestPayState and requestPayReason

✓ Success Response:

```

HTTP/1.1 202 Accepted

{
  "requestPayRef": "003-oYVl3qcZdm4Z3",
  "paymentRef": "tyhkfffhnjy2345",
  "paymentDescription": "request to pay your bill",
  "paymentOption": "EXACT",
  "paymentDueDate": "2020-12-14 10:45:44",
  "amount": {
    "amount": "2300",
    "currency": "TZS"
  },
  "payee": {
    "identifierType": "MSISDN",
    "identifier": "255621804189",
    "fspId": "003",
    "fullName": "MAGANGA SUPPLIES",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19501007111010000126"
    }
  },
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "004",
    "fullName": "JOHN JOHN",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    }
  },
  "requestPayState": "INITIATED",
  "requestPayStateReason": "Successfully received by TIPS",
  "createdDate": "2020-10-14 10:45:46"
}

```

✓ Error response:

```

HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}

```

✓ **Description of variables:**

Variable	Description and Examples
URL	/messageRequestPay/{requestPayRef}
Headers	Described in headers section 3.2
paymentDueDate	This is the date the request to pay expires, after that date no payments are allowed to that request to pay.
requestPayState	The state the request to pay to be updated to. Possible values are: • CANCELLED – The payee FSP cancels the request to pay. • REJECTED – the payer FSP rejects the request to pay. NOTE: Only the payee FSP can cancel. Only the payer FSP can reject.
requestPayReason	The reason to cancel or reject the request to pay.

4.4.3 GET /messageRequestPay/{requestPayRef}

✓ **Description:** This request is used for the enquiry of request to pay and associated payments if any.

✓ **Implementation:** Implemented by TIPS and consumed by FSPs.

✓ **Request:**

```
GET /messageRequestPay/{requestPayRef} HTTP/1.1
```

```
Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "004"
```

✓ **Success Response:**

```
HTTP/1.1 200 OK
{
  "requestPayRef": "003-af81bf4a-eff5-45a9-aabe-13fec34933ad",
  "paymentRef": "9dc8293e-cb56-472a-a655-93d78e2dfaf4",
  "paymentDescription": "string",
  "paymentOption": "PARTIAL",
  "paymentDueDate": "2023-07-30 22:20:00",
  "amount": {
    "amount": "55000.00",
    "currency": "TZS"
  },
  "payee": {
    "identifierType": "MSISDN",
    "identifier": "255621804189",
    "fspId": "003",
    "fullName": "MAGANGA SUPPLIES",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
```

```

        "type": "NIN",
        "value": "19501007111010000126"
    },
    "payer": {
        "identifierType": "MSISDN",
        "identifier": "255556789922",
        "fspId": "004",
        "fullName": "JOHN JOHN",
        "accountCategory": "PERSON",
        "accountType": "WALLET",
        "identity": {
            "type": "NIN",
            "value": "19831007175010005135"
        }
    },
    "requestPayState": "PARTIAL_PAID",
    "requestPayStateReason": "The request to pay is partially paid",
    "createdDate": "2023-04-03 10:45:03",
    "totalPayments": "5700.00",
    "associatedTransfers": [
        {
            "payerRef": "002-05dc1bf1-4d36-48d7-9bc6-a55d00d9f0e8",
            "payeeRef": null,
            "switchRef": "c4d29c4e-7540-4f60-8f10-d3f4177a3b0a",
            "amount": {
                "amount": "8000.00",
                "currency": "TZS"
            },
            "transferDate": "2023-04-03 10:50:12",
            "transferCompletedDate": "",
            "transferState": "RESERVED",
            "payer": {
                "fspId": "002",
                "fullName": "BEATRICE HAMISI",
                "accountCategory": "PERSON",
                "accountType": "WALLET",
                "identity": {
                    "type": "NIN",
                    "value": ""
                }
            }
        },
        {
            "payerRef": "002-abd13daf-eee6-475a-8260-09d181bf611e",
            "payeeRef": "003-32e4628f-8efd-4edd-97b9-c7f043c8426e",
            "switchRef": "8f4b45aa-44ee-4645-8e5d-33aef90d8875",
            "amount": {
                "amount": "5700.00",
                "currency": "TZS"
            },
            "transferDate": "2023-04-03 10:48:59",
            "transferCompletedDate": "2023-04-03 10:49:50",
            "transferState": "COMMITTED",
            "payer": {
                "fspId": "002",
                "fullName": "BEATRICE HAMISI",
                "accountCategory": "PERSON",
                "accountType": "WALLET",
                "identity": {
                    "type": "NIN",
                    "value": ""
                }
            }
        }
    ]
}

```

✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

4.5 Settlement API

4.5.1 POST /settlements

- ✓ **Description:** TIPS use this request to notify the FSPs on their summary of settlement position as at closure of every clearing window.
- ✓ **NOTE:** The TIPS portal will also provide a facility for FSPs to access and download details of individual transfers in batches for reconciliation if need be.
- ✓ **Implementation:** Implemented by FSPs and consumed by TIPS
- ✓ **Request:**

```
POST /settlements HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "tips"
Fsp-destination: "003"

{
  "settlementWindow": {
    "id": "2",
    "openingDate": "2022-06-13 16:49:11",
    "closingDate": "2022-06-13 16:58:11",
    "description": "window for 14 Oct 2020"
  },
  "outgoingTransactions": [
    {
      "currency": "TZS",
      "volume": "500",
      "value": "25000000.00"
    }
  ],
  "incomingTransactions": [
    {
      "currency": "TZS",
      "volume": "300",
      "value": "10000000.00"
    }
  ],
  "netTransactions": [
    {
      "currency": "TZS",
      "volume": "-200",
      "value": "-15000000.00"
    }
  ],
  "accountsPosition": [
    {
      "accountType": "POSITION",
      "currency": "TZS",
      "amount": "60500000.00",
      "ledgerType": [
        {
          name: "PRINCIPAL",
          amount: "60500000.00"
        }
      ]
    }
  ]
}
```

```

    }
  ],
  {
    "accountType": "FEES",
    "currency": "TZS",
    "amount": "-12500000.00",
    "ledgerType": [
      {
        name: "INTERCHANGE_FEE",
        amount: "-10500000.00"
      },
      {
        name: "PROCESSING_FEE",
        amount: "-2000000.00"
      }
    ]
  }
]
}

```

✓ Success Response:

```
HTTP/1.1 200 OK
```

✓ Error response:

```
HTTP/1.1 400 Bad Request
```

```

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}

```

✓ Description of variables:

Variable	Description and Examples
URL	/settlements
Headers	Described in request headers section 3.2
settlementWindow	This object defines the settlement/clearing window that has just been closed in TIPS. It indicates the unique ID as well as opening and closing date and time of the window.
outgoingTransactions	This object defines the summary of COMMITTED transactions that the FSP participated as a payer (sending) FSP. • Currency: this is the currency of the transactions. • Volume: This is the total number of transactions • Value: This is the total value of transactions.
incomingTransactions	This object defines the summary of COMMITTED transactions that the FSP participated as a payee (receiving) FSP.

	• Currency: this is the currency of the transactions. • Volume: This is the total number of transactions Value: This is the total value of transactions.
netTransactions	This object defines the net transactions. It will be the difference (incoming – outgoing) of incoming and outgoing transactions (excluding funds in and funds out) in a particular settlement window.
accountsPosition	This object denoted the closing ledger balance in TIPS as at the closure of a specified settlement window in TIPS. • AccountType: This is the type of ledger account in TIPS. There are of two types. POSITION (to provision principal amount of the transfers) and FEES (to provision of the fees associated with the transfers) • Currency: The currency of the ledger account. • Amount: The ledger account actual balance.

4.5.2 Reconciliation file – COMPLETE

- ✓ **Description:** TIPS generates CSV file for completed (COMMITTED) transfers for FSP for each clearing window. This file is made available for FSPs to download from TIPS FTP server after the successful closure of that clearing window.

✓ **Description of fields:**

Column name	Description
transactionDate	Data and time the transaction is received at TIPS
transactionCompletedDate	Date and time the transaction completed in TIPS
settlementWindowDate	Date of settlement window, used for reconciliation
payerFSP	The name of payer FSP
payerFullName	The name of payer customer
payerIdentifierType	The identifier type of payer customer
payerIdentifier	The identifier of the payer customer
payeeFSP	The name of payee FSP
payeeFullName	The name of payee customer
payeeIdentifierType	The identifier type of payee customer
payeeIdentifier	The identifier of the payee customer

payerRef	Payer FSP transaction reference
payeeRef	Payee FSP transaction reference
switchRef	TIPS transaction reference
amount	Transaction amount
balance	Balance/position after the transaction
currency	Currency of the transaction amount
settlementWindowId	The unique ID of the settlement window in which the transfer completed
scenario	The scenario of a transaction record. The values can be OPENING_BALANCE, TRANSFER, FUNDS_IN, FUNDS_OUT, TRANSFER_REVERSAL, PAYMENT, PAYMENT_REQUESTPAY, TRANSFER_IMT or CLOSING_BALANCE
transactionCategory	Category of the transaction
direction	Direction of transaction. The values can be IN or OUT
transactionState	The final state of the transaction. The values can be COMMITTED
transactionStateReason	The transaction is successfully completed and committed

4.5.3 Reconciliation file – INCOMPLETE

- ✓ **Description:** TIPS generates CSV file for incomplete (ABORTED, INVALID and all pending) transfers for FSP for each clearing window. This file is made available for FSPs to download from TIPS FTP server after the closure of that clearing window.

- ✓ **Description of fields:**

Column name	Description
transactionDate	Data and time the transaction is received at TIPS
transactionCompletedDate	Date and time the transaction completed in TIPS
settlementWindowDate	Date of settlement window, used for reconciliation
payerFSP	The name of payer FSP
payerFullName	The name of payer customer
payerIdentifierType	The identifier type of payer customer
payerIdentifier	The identifier of the payer customer

payeeFSP	The name of payee FSP
payeeFullName	The name of payee customer
payeeIdentifierType	The identifier type of payee customer
payeeIdentifier	The identifier of the payee customer
payerRef	Payer FSP transaction reference
payeeRef	Payee FSP transaction reference
switchRef	TIPS transaction reference
amount	Transaction amount
balance	Balance not populated for incomplete transfers
currency	Currency of the transaction amount
settlementWindowId	The unique ID of the settlement window in which the transfer completed
scenario	The scenario of a transaction record. The values can be OPENING_BALANCE, TRANSFER, FUNDS_IN, FUNDS_OUT, TRANSFER_REVERSAL, PAYMENT, PAYMENT_REQUESTPAY, TRANSFER_IMT or CLOSING_BALANCE
transactionCategory	Category of the transaction
direction	Direction of transaction. The values can be IN or OUT
transactionState	The current state of the transaction.
transactionStateReason	The state description of a transfer

4.6 Fraud API

4.6.1 POST /fraud

- ✓ **Description:** This request is used by FSPs to register materialized fraud cases to the central fraud register in TIPS.
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs.
- ✓ **Request:**

```
POST /fraud HTTP/1.1
Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "TIPS"

{
  "fspId": "string",
```

```

    "identifierType": "string",
    "identifier": "string",
    "fullName": "string",
    "identityType": "string",
    "identityValue": "string",
    "reasons": "string",
    "status": "BLACKLISTED"
  }

```

✓ Success Response:

```

HTTP/1.1 201 Created

{
  "fraudRegisterId": 0,
  "participantId": 0,
  "fspId": 0,
  "identifierType": "string",
  "identifier": "string",
  "fullName": "string",
  "identityType": "string",
  "identityValue": "string",
  "reasons": "string",
  "status": "string",
  "createdDate": "string"
}

```

✓ Error response:

```

HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}

```

✓ Description of variables:

Variable	Description and Examples
URL	/fraud
Headers	Described in request headers section 3.2
fspId	The ID of the FSP owning the identifier. Each FSP will have a unique 3 characters code identification.
identifierType	The type of the identifier. Possible values for this request are: • BUSINESS – identify merchants • ALIAS – identify TIPS identifiers • MSISDN – identify mobile phone numbers • BANK – identify bank account numbers • WALLET – identifies wallets that are not identified by MSISDNs
identifier	The account identifier. Examples values for this request are: • BUSINESS – e.g. 00300125

	• ALIAS – e.g. TANAPA • MSISDN – e.g. 255621804189 • BANK – e.g. 01258796325 • WALLET – e.g. 10725542585
fullName	The full name of the account owner.
identityType	The identification of the account owner. • The “type” field can be “NIN” for National ID, while the “value” is the ID number itself without hyphens. • The “type” field can be “TIN” for Tax Id Numbers, while the “value” is the ID number itself without hyphens.
identityValue	The identification of the account owner. • The “type” field can be “NIN” for National ID, while the “value” is the ID number itself without hyphens. • The “type” field can be “TIN” for Tax Id Numbers, while the “value” is the ID number itself without hyphens.
reasons	Reason for adding the party into fraud register
status	The status when added to the fraud register. Possible value is BLACKLISTED or FLAGED

4.6.2 PUT /fraud/{id}

- ✓ **Description:** This request is used by FSPs to update status of the fraud cases they have registered in the central fraud register.
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs.
- ✓ **Request:**

```
PUT /fraud/{id} HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "TIPS"

{
  "fspId": "string",
  "status": "BLACKLISTED",
  "reasons": "string"
}
```

- ✓ **Success Response:**

```
HTTP/1.1 200 OK

{
  "fspId": "string",
  "status": "BLACKLISTED",
  "reasons": "string"
}
```

✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/fraud/{id}
Headers	Described in request headers section 3.2
fspId	The ID of the FSP owning the identifier. Each FSP will have a unique 3 characters code identification.
reasons	Reason for updating the party into fraud register
status	The status when added to the fraud register. Possible value is BLACKLISTED or WHITELISTED or FLAGED

4.6.3 GET /fraud

- ✓ **Description:** This request is used to enquire details of the fraud cases registered in central fraud register in TIPS.
- ✓ **Implementation:** Implemented by TIPS and consumed by FSPs.
- ✓ **Request:**

```
GET /fraud HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "tips"
```

✓ **Success Response:**

```
HTTP/1.1 200 OK

[
  {
    "fraudRegisterId": 0,
    "fspId": "string",
    "fullName": "string",
    "identifierType": "string",
    "identifier": "string",
    "identityType": "string",
    "identityValue": "string",
    "reason": "string",
    "status": "string",
    "createdDate": "string"
  }
]
```

✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/fraud
Headers	Described in request headers section 3.2
fspId	The ID of the FSP owning the identifier. Each FSP will have a unique 3 characters code identification.
identifierType	The type of the identifier. Possible values for this request are: • BUSINESS – identify business or merchants • ALIAS – identify TIPS identifiers • MSISDN – identify mobile phone numbers • BANK – identify bank account numbers • WALLET – identifies wallets that are not identified by MSISDNs
identifier	The account identifier. Examples values for this request are: • BUSINESS – e.g. 00300125 • ALIAS – e.g. TANAPA • MSISDN – e.g. 255621804189 • BANK – e.g. 01258796325 • WALLET – e.g. 10725542585
fullName	The full name of the account owner.
identityType	The identification of the account owner. • The “type” field can be “NIN” for National ID, while the “value” is the ID number itself without hyphens. • The “type” field can be “TIN” for Tax Id Numbers, while the “value” is the ID number itself without hyphens.
identityValue	The identification of the account owner. • The “type” field can be “NIN” for National ID, while the “value” is the ID number itself without hyphens.

	The “type” field can be “TIN” for Tax Id Numbers, while the “value” is the ID number itself without hyphens.
reasons	Reason for adding the party into fraud register
status	The status in the fraud register

4.7 Communication Messaging API

4.7.1 POST /message

- ✓ **Description:** This request is used to send communication messages. TIPS will also use this request to broadcast messages to the FSPs.
- ✓ **Implementation:** Implemented by both TIPS and FSPs and consumed by both TIPS and FSPs.
- ✓ **Request:**

```
POST /message HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "003"
Fsp-destination: "004"
{
  "sender": {
    "reference": "string",
    "fspId": "string",
    "user": "string"
  },
  "recipients": [
    {
      "fspId": "string",
      "user": "string"
    }
  ],
  "subject": "string",
  "body": "string",
  "notificationType": "string"
}
```

- ✓ **Success Response:**

```
HTTP/1.1 200 OK
```

- ✓ **Error response:**

```
HTTP/1.1 400 Bad Request

{
  "errorInformation": {
    "errorCode": "number",
    "errorDescription": "string"
  }
}
```

✓ **Description of variables:**

Variable	Description and Examples
URL	/message
Headers	Described in request headers section 3.2
sender	This is the details of the sender of the communication. It is the FSP ID and the reference of the sender.
recipients	The recipient of the message, this is the 3 character FSP ID.
subject	The subject of the communication.
body	The message contents.
notificationType	This is the type of notification for communication message. Possible values are: COMMUNICATION_MESSAGE

4.8 API Implementation summary

No	API service	API usage	TIPS		FSP	
			Implement	Consume	Implement	Consume
Lookup API						
1	GET /parties/{identifierType}/{identifier}/sync	Name lookup	✓	✓	✓	✓
2	POST /participants/{identifierType}/sync	Registration of TIPS ALIAS	✓			✓
3	PUT /participants/{identifierType}/{identifier}/sync	Update of TIPS ALIAS	✓			✓
4	GET /participants/{identifierType}/sync	Retrieval of TIPS ALIAS	✓			✓
Transfer API						
5	POST /transfers	Send transfers/payments	✓	✓	✓	✓
6	PUT /transfers/{payerRef} (for confirmation)	Payee FSP confirm transfers/payments	✓			✓
7	PUT /transfers/{payerRef} (for notification)	Success callback for transfer/payments		✓	✓	
8	PUT /transfers/{payerRef}/error	Error callback for transfer/payments		✓	✓	
9	GET /transfers/{payerRef}	TIPS transaction enquiry	✓			✓
10	GET /transfers/{payerRef} (for triggering reminders)	FSP transaction enquiry		✓	✓	
Transfer Reversal API						
11	POST /messageTransfersReversal	Initiate transfer/payment reversal	✓	✓	✓	✓
12	PUT /messageTransfersReversal{payerRef}	Update & notify reversal messages	✓	✓	✓	✓
13	GET /messageTransfersReversal{payerRef}	Enquire reversal messages	✓			✓
14	POST /transfersReversal	Reverse a transfer/payment	✓			✓
15	PUT /transfersReversal/{payerRef}	Success callback for reversal		✓	✓	
16	PUT /transfersReversal/{payerRef}/error	Error callback for reversal		✓	✓	
17	GET /transfersReversal/{payerRef}	Enquiry of reversals	✓			✓

Request to Pay API						
18	POST /messageRequestPay	Initiate request to pay	✓	✓	✓	✓
19	PUT /messageRequestPay/{requestPayRef}	Update and notify request to pay	✓	✓	✓	✓
20	GET /messageRequestPay/{requestPayRef}	Enquiry of request to pay	✓			✓
Settlement API						
21	POST /settlements	Notification of settlement positions		✓	✓	
Fraud API						
22	POST /fraud	Registration of fraud cases	✓			✓
23	PUT /fraud/{id}	Update of fraud cases	✓			✓
24	GET /fraud	Retrieval of fraud cases	✓			✓
Communication Messaging API						
25	POST /message	Send communication message	✓	✓	✓	✓

5. Error Handling

This section provides list of errors implemented in TIPS. The errors inform reasons for failure of processing TIPS transactions.

Code	Name	Description
COMMUNICATION ERRORS		
1000	Communication error	Generic communication error.
1001	Destination communication error	Destination of the request failed to be reached. Indicates that a Peer FSP is unreachable.
SERVER ERRORS		
2000	Internal server error	Generic unexpected exception. Indicates a bug or unhandled error case.
2001	Not implemented	Service requested is not supported by the server
2002	Service currently unavailable	Service requested is temporarily unavailable.
2003	Server timed out	Timeout has occurred because something took longer than expected.
2004	Server busy	Server is rejecting requests due to overloading.
CLIENT ERRORS		
3000	Generic client error	Used in order not to disclose private information.
3001	Unacceptable version requested	Client requested to use un supported protocol version.
3003	Add Party information error	Error occurred while adding or updating a Party.
3100	Generic validation error	Generic validation error of the request.
3101	Malformed syntax	Format of the parameter is not valid.
3102	Missing mandatory element	Mandatory element in the data model is missing.
3103	Too many elements	Number of elements of an array exceeded.

3104	Too large payload	Size of the payload exceeds the maximum size.
3105	Invalid signature	Some parameters have changed in the message, making the signature invalid.
3106	Modified request	Request with the same ID has previously been processed.
3107	FSP-ID mismatch	Header and payload source and destination FSP-ID does not match.
3200	Timeout error	Processing time exceed configurable timeout
3201	Destination FSP error	Destination FSP does not exist.
3202	Payer FSP ID not found	Provided Payer FSP ID not found.
3203	Payee FSP ID not found	Provided Payee FSP ID not found.
3204	Party not found	Party with provided identifier, identifier type not found.
3208	Transfer Ref not found	Provided Transfer Ref. was not found on the server.
3300	Transaction request expired	Client requested to use an expired transaction request.
PAYER ERRORS		
4000	Generic Payer error	Generic error related to the Payer or Payer FSP.
4001	Payer FSP insufficient liquidity	Payer FSP has insufficient liquidity to do the transfer.
4002	Payer FSP unsupported transaction type	Payer FSP does not support or has rejected the requested transaction type.
4003	Payer unsupported currency	Payer does not have an account that supports the requested currency.
4004	Payer limit error	Payer is making more transfers or payments than they are allowed.
4005	Payer permission error	Payer or Payer FSP does not have the access rights to perform the service.
4006	Payer duplicate reference	Transfer request from payer with duplicate reference

PAYEE ERRORS		
5000	Generic Payee error	Generic error related to the Payer or Payer FSP.
5001	Payee FSP insufficient liquidity	Payee FSP has insufficient liquidity to do the transfer.
5002	Payee FSP unsupported transaction type	Payee FSP does not support or has rejected the requested transaction type.
5003	Payee unsupported currency	Payee does not have an account that supports the requested currency.
5004	Payee limit error	Payee is receiving more transfers than they are allowed.
5005	Payee permission error	Payee or Payee FSP does not have the access rights to perform the service.
5006	Payee duplicate reference	Transfer request from payee with duplicate reference

6. Appendix 1: Operations Guideline

The operations guide provides a guidance on how to perform various operations on the TIPS platform. The purpose is to standardize the integration and operationalization among FSPs. For each operation, this guide:

- ✓ Describes a service offered,
- ✓ Explains sequence of activities and identifies the API resource(s) or TIPS Portal step(s) required to accomplish a service, and
- ✓ Provides outcome possibilities, interpretation and next steps of actions required to accomplish a service.

Appendix 1.1: Guide to perform lookup-out

The payer FSP performs a lookup to TIPS so that the request is communicated to payee FSP and the payee customer details are returned.

Lookup-out is done by the payer FSP consuming the TIPS API resource as specified in section 4.1.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When the payee customer details are returned successfully to payer FSP as specified in the respective success response.
 - Interpretation: The lookup-out is successful. The account/identifier is valid.
 - Action: The payer FSP receives the payee customer account details and uses them to proceed with transfer or payment.
- ✓ **FAILURE:** When the payee customer details are not returned to payer FSP. The failure can be caused by a number of factors such as incorrect request, identifier not found at payee FSP, network problems or processing errors. TIPS responds to payer FSP with details of failure as specified in the respective error response.
 - Interpretation: The lookup-out is not successful.
 - Action: The payer FSP should verify and modify the request (if need be) or check network connectivity and try again.
 - **NOTE:** The payer FSP should not proceed with transfer if lookup-out is not successful.

Appendix 1.2: Guide to perform lookup-in

TIPS performs a lookup to the payee FSP so that the payee customer account details are returned so that are communicated back to the payer FSP.

Lookup-in is done by TIPS consuming the payee FSP API resource as specified in section 4.1.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When identifier details are returned successfully to TIPS by the payee FSP as specified in the respective success response.
 - Interpretation: The lookup-in is successful. The account/identifier is valid.
 - Action: TIPS receives the payee customer details and subsequently communicates it to the payer FSP.
- ✓ **FAILURE:** When payee customer details are not returned to TIPS. The failure can be caused by a number of factors such as incorrect request, identifier not found at payee FSP, network problems or processing errors. The payee FSP should respond to TIPS on failure encountered as specified in the respective error response.
 - Interpretation: The lookup-in is not successful.
 - Action: TIPS receives the error response from payee FSP (or terminates request in case of network problems) and communicates the error back to the payer FSP.

Appendix 1.3: Guide to perform transfer-out

The payer FSP is sending outgoing transfers or payments.

The transfer-out process is for the payer FSP and is accomplished in three steps as follows:

- ✓ Step 1: Initiation
- ✓ Step 2: Callback
- ✓ Step 3: Enquiry (only when need be)

Step 1: Initiation: Transfer-out initiation is done by payer FSP consuming the TIPS API resource as specified in section 4.2.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The transfer-out initiation is successful. The transaction is successfully received by TIPS and its state changes to RECEIVED.
 - Action: The payer FSP waits for the request to be processed and will be notified of the outcome via a callback (step 2: callback) which under normal circumstances should be in near real-time.
 - **NOTE:** The payer FSP should hold/reserve funds from the payer customer's account until the callback is received as there is no guarantee that the transaction will complete successfully.
- ✓ **FAILURE:** When TIPS does not accept the request. The failure can be caused by a number of factors such as incorrect request, network problems or processing errors. TIPS responds to payer FSP with details of failure encountered as specified in the respective error response.
 - Interpretation: The transfer-out initiation is not successful.
 - Action: The payer FSP should verify and modify the request based on the error message or check network connectivity and try again.
 - **NOTE:** For cases of network failure/timeout the payer FSP is advised to retry after approximately 5 minutes and if possible do an enquiry to TIPS first to verify if the request didn't reach before re-sending to avoid sending duplicate transfers or payments. Duplicate transfers with the same payer reference will be rejected.

Step 2: Callback: Transfer-out callback notifies the payer FSP on the processing outcome of a transfer-out.

NOTE:

- a. No intermediate callback(s) are sent during processing until completion.
- b. In circumstances that the payer FSP does not receive callback within SLA time, then there are possibilities that 1) there are processing delays at payee FSP or 2) there are processing delays at TIPS or 3) TIPS failed to deliver the callback to payer FSP due to network problems or other problems. Therefore, the payer FSP can proceed with

- transfer enquiry to TIPS (step 3) so as to know the status of the transfer and act accordingly.
- c. It is recommended to start enquiry (step 3) at least 5 minutes after sending the transfer-out in case the callback is not received or transfer not finalized.

The payer FSP will receive one of the following two possible callbacks depending on completion status of a transfer.

- i. **Success callback:** to notify successful completion of a transfer-out.

This is done by TIPS consuming the payer FSP API resource as specified in section 4.2.4

- Interpretation: The transfer-out has been successfully processed (by TIPS and payee FSP). The TIPS settlement account for payer FSP is debited and that for payee FSP is credited. The transfer state changes to COMMITTED.

- **NOTE:** It also implies that the payee customer has received funds.

- Action: The payer FSP should complete the transaction on their side and notify their customers accordingly.

- ii. **Unsuccessful callback:** to notify unsuccessful completion of a transfer-out.

This is done by TIPS consuming the payer FSP API resource specified in section 4.2.5

- Interpretation: The transfer-out is not successful. The transfer state changes to INVALID or ABORTED. No further processing of the transfer.
- Action: The payer FSP should roll back the transfer and give an opportunity to their customers to initiate a new transfer (with a different payer reference).

Step 3: Enquiry: This step is optional. To be executed when callback (step 2 above) is not received within SLA time or anytime when the payer or payee FSP wants to know the details of the transfer. The procedure is explained in Appendix 1.5

Appendix 1.4: Guide to perform transfer-in

The payee FSP is receiving incoming transfers or payments.

The transfer-in process is for the payee FSP and is accomplished in four steps as follows:

- ✓ Step 1: Initiation
- ✓ Step 2: Confirmation reminder (only when need be)
- ✓ Step 3: Payee FSP confirmation of credit or no-credit
- ✓ Step 4: Callback

Step 1: Initiation: Transfer-in initiation is done by TIPS consuming the payee FSP API resource as specified in section 4.2.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payee FSP accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The transfer-in initiation is successful. The transfer is successfully received by payee FSP and its state is ACKNOWLEDGED (in some cases RESERVED).
 - Action: TIPS waits for the payee FSP to process the transfer-in request and should subsequently send an appropriate confirmation to TIPS (step 3)
 - **NOTE:** The payee FSP should perform all validations, process and complete the transfer (including crediting beneficiary) before sending a confirmation to TIPS.
- ✓ **FAILURE:** When the transfer-in request has not been delivered to payee FSP. Failure can be due to network or bad request.
 - Interpretation: The transfer-in initiation is not successful. The transfer state remains as RESERVED.
 - Action: TIPS marks this transfer-in request as undelivered to payee FSP and TIPS will trigger the confirmation reminders (step2) in case no confirmation (step 3) is received from payee FSP within SLA time.

Step 2: Confirmation reminder: This is optional step to be performed IF AND ONLY IF no confirmation (step 3) is received from payee FSP within SLA time. TIPS will do a confirmation reminder to payee FSP to trigger finalization of the transfer or payment request. This is done by TIPS sending an enquiry to the payee FSP API as specified in section 4.2.3. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payee FSP accepts the enquiry request and responds successfully as specified in the respective success response.
 - Interpretation: The confirmation reminder to payee FSP is successful.
 - Action: The payee FSP should immediately process and send an appropriate confirmation (step 3) that instructs TIPS to complete the transfer or payment.

- ✓ **FAILURE:** When the enquiry fails and TIPS receives errors as specified in the respective error response or the payee FSP does not respond to the enquiry. Failure can be due to network or bad request or payee FSP errors.
 - Interpretation: The confirmation reminder to payee FSP is not successful.
 - Action: TIPS continues to marks the transfer-in request as undelivered and TIPS will perform another reminder incase no confirmation (step 3) is received from payee FSP.
 - **NOTE:** TIPS will do up to 5 confirmation reminders in an interval of 6 minutes each in case no confirmation is received. After that time, it marks the transaction as pending, thereafter, manual intervention will be required. Transfer state remains RESERVED.

Step 3: Confirmation: Transfer-in confirmation is done by payee FSP consuming the TIPS API resource as specified in section 4.2.2. The confirmation is to instruct TIPS to COMMIT or ABORT a transfer as appropriate. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The confirmation is successful. The transfer state changes to state CONFIRMED.
 - Action: Payee FSP waits for TIPS to process the transfer and will be notified on the outcome via a callback (step 4).

- ✓ **FAILURE:** When TIPS does not accept the confirmation request. Failure can be due to network or bad request.

- Interpretation: The transfers-in confirmation is not successful. The transfer state remains as ACKNOWLEDGED (or in some cases RESERVED)
- Action: The payee FSP should modify request (if need be) and is required to send the confirmation again.

NOTE: The confirmation process can also be accomplished through the TIPS Portal in case the payee FSP has challenges or issues of sending the confirmation from their platforms. Refer to the TIPS Portal user guide.

Step 4: Callback: Transfer-in callback notifies the payee FSP on the processing outcome the transfer. This callback can be final (commit or abort) or not final (error) in processing the request.

The payee FSP will receive one of the following three possible callbacks depending on processing status of a transfer:

- i. **Commit callback:** to notify successful COMMIT of a transfer in TIPS.

This is done by TIPS consuming the payee FSP API resource as specified in section 4.2.4.

- Interpretation: The transfer has been successfully processed in TIPS. The TIPS settlement account for payer FSP is debited and that for payee FSP is credited. The transfer state changes to COMMITTED.
- Action: The payee FSP marks this transaction as completed and settled in TIPS.

- ii. **Abort callback:** to notify successful ABORT of a transfer

This is done by TIPS consuming the payee FSP API resource as specified in section 4.2.5 with specific description of the error.

- Interpretation: TIPS has aborted the transfer as per the instructions from payee FSP. The transaction state changes to ABORTED.
- Action: The payee FSP marks this transaction as completed and that it has not affected their TIPS settlement account.

- iii. **Error callback:** to notify errors occurred during processing

There are errors regarding issues with the confirmation request (e.g. invalid signature, duplicate payee references, invalid date and others). TIPS has failed to

validate and process the confirmation request and sends the error callback by consuming the payee FSP API resource as specified in section 4.2.5 with specific description of the error.

- Interpretation: The transfer is not finalized. The transfer remains in state of ACKNOWLEDGED (or in some cases RESERVED).
- Action: The payee FSP should modify the request and is required to re-send the confirmation (step 3) again.

Appendix 1.5: Guide to perform transfer enquiry

Transfer enquiry enables a payer or payee FSP to request information of a particular TIPS transfer. The requesting FSP must have participated in that transfer as either payer or payee, otherwise the enquiry shall not be served.

Transfer-enquiry is done by an FSP consuming the TIPS API resource as specified in section 4.2.6. The outcome for this can be as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response or error response with error code 3202: Transfer not found (meaning the transfer with the provided reference is not found/processed in TIPS)
 - Interpretation: The request is received and processed successfully at TIPS and returns a response. Sure of existence or non-existence of a transfer in TIPS.
 - Action:

Response & Interpretation	FSP Action
Enquiry returns transfer state (the value of parameter transferState) equal to [COMMITTED]. Means the transfer has reached terminal state, completed and settled successfully in TIPS.	Mark transfer as complete and commit.
Enquiry returns transfer states (the value of parameter transferState) equals in [INVALID, ABORTED]. Means the transfer has reached terminal state but is unsuccessful and has not affected the accounts.	Mark transfer as unsuccessful and rollback.

Enquiry returns error response that “transfer is not found” with error code 3202 as follows: <pre> {"errorInformation": {"errorCode":3202,"errorDescription":"Provided Payer reference or payer FSP ID was not found."} } </pre>	Mark transfer as not received or processed at TIPS. The FSPs should also consider this transfer as non-existence in TIPS.
Enquiry returns transfer states (the value of parameter transferState) equals in [RECEIVED, VALIDATED, RESERVED, ACKNOWLEDGED, CONFIRMED]. Means the transfer is still under processing and it will be finalized later.	Mark the transfer as still processing. Should continue to wait and enquire again later.
Enquiry returns all other errors (other than error code 3202). Means the request has encountered errors and has failed to return a response as expected.	Check the request and modify as required and enquire again.

- ✓ **FAILURE:** When response is error as specified in respective error response for all errors other than 3202: “Transfer not found”
 - Interpretation: The enquiry request is not successful.
 - Action: The FSP should verify and modify the request based on the error information or check network connectivity and try again.

Appendix 1.6: Guide to perform reversal-out

The payer FSP is reversing a transfer.

The reversal-out is for the original payer FSP only and is accomplished in three steps as follows:

- ✓ Step 1: Initiation
- ✓ Step 2: Update callback
- ✓ Step 3: Reversal callback

Step 1: Initiation: Reversal-out initiation is done by payer FSP consuming the TIPS API resource as specified in section 4.3.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The reversal-out initiation is successful. The state of transfer reversal request changes to INITIATED.
 - Action: The payer FSP waits for the reversal request to be processed and will be notified via a callback (step 2: update callback).
- ✓ **FAILURE:** When TIPS does not accept the request. The failure can be caused by a number of factors such as network problems, incorrect request, processing errors. TIPS responds to payer FSP with details of failure encountered as specified in the respective error response.
 - Interpretation: The reversal-out initiation is not successful.
 - Action: The payer FSP should verify and modify the request based on the error message or check network connectivity and try again.

Step 2: Update callback: This callback notifies the payer FSP on the processing status of their reversal-out request as updated by the payee FSP. This is done by TIPS consuming the payer FSP API resource as specified in section 4.3.2. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payer FSP accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The update callback is successful. At this point, the transfer reversal request can be updated into any of the following states CONFIRMED_HOLD, CONFIRMED_WITHDRAW or CANCELLED.
 - Action: The payer FSP can perform different actions based on the response state as follows.

Response & Interpretation	FSP Action
If the state is CONFIRMED_HOLD , this means the payee FSP has hold funds from their customer account and upon confirmation with their customer, they will subsequently return the funds.	The payer FSP waits for another callback. This can either be another update callback (step 2) with state CANCELLED or reversal callback (step 3). Upon receipt of this callback, the payer FSP can notify their customer that the funds have been hold and will be reversed upon confirmation.
If the state is CONFIRMED_WITHDRAW , this means the payee FSP confirms that their customer has already withdrawn the funds.	The payee FSP confirms that their customer has withdrawn the funds. The payer FSP should notify their customer that the funds have already been withdrawn. The reversal is still possible if the payee customer funds in their account.
If the state is CANCELLED , means that payee FSP has rejected the reversal request for the reasons specified.	The payer FSP should consider this reversal request as impossible to proceed further. NOTE: The payer FSP has an opportunity to initiate a new reversal request for the same transfer if there is still need to do so.

- ✓ **FAILURE:** When TIPS fails to deliver the callback or payer FSP fails to accept the callback due to network problems or incorrect request respectively.
 - Interpretation: The payer FSP is not updated on the reversal status.
 - Action: TIPS will try to re-send the update callback upon request. Also, the payer FSP can perform reversal message enquiry process as guided in Appendix 1.8 and use the response to update the status on their end.

Step 3: Reversal callback: This callback notifies the payer FSP on the successful completion of the transfer reversal process. This is done by TIPS consuming the payer FSP API resource as specified in section 4.3.5. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payer FSP accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The reversal-out is successful and completed. The state of transfer reversal changes to REVERSED and the reversal transfer is performed. The original payer FSP settlement account in TIPS has been credited and original payee FSP account debited.
 - Action: The original payer FSP should immediately return the funds to their customer account and marks this reversal as completed.
- ✓ **FAILURE:** When TIPS fails to deliver the callback or payer FSP fails to accept the callback due to network problems or incorrect request respectively.
 - Interpretation: The payer FSP is not updated on the reversal status.
 - Action: TIPS will try to re-send the reversal callback upon request. Also, the payer FSP can perform reversal message enquiry process as guided in Appendix 1.8 and use the response to update the status on their end.

Appendix 1.7: Guide to perform reversal-in

The payee FSP is processing incoming reversal requests and performing actual reversal.

The reversal-in procedure is for the payee FSP only and is accomplished in four steps as follows:

- ✓ Step 1: Initiation
- ✓ Step 2: Update confirmation
- ✓ Step 3: Reversal
- ✓ Step 4: Reversal callback

Step 1: Initiation: Reversal-in initiation is done by TIPS consuming the payee FSP API resource as specified in section 4.3.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payee FSP accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The reversal-in initiation is successful. The state of transfer reversal request changes to RECEIVED.
 - Action: TIPS waits for the reversal request to be processed by payee FSP and will be updated on outcome via a callback (step 2: update confirmation).
- ✓ **FAILURE:** When TIPS fails to deliver the reversal request. The failure can be caused by a number of factors such as network problems, incorrect request. On errors, payee FSP responds to TIPS with details of failure encountered as specified in the respective error response.
 - Interpretation: The reversal-in initiation is not successful. The reversal request remains in state INITIATED.
 - Action: TIPS will re-send or remind the payee FSP on the pending reversal request. Also, the payee FSP can enquire their pending reversal-in requests and use the response to process on their end.

Step 2: Update confirmation: reversal-in update confirmation is done by payee FSP consuming the TIPS API resource as specified in section 4.3.2. The payee FSP uses the update confirmation to update TIPS whether they have CONFIRMED_HOLD, CONFIRMED_WITHDRAW or CANCELLED the reversal request. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The update confirmation is successful. The state of reversal request is updated to any of the following states: CONFIRMED_HOLD, CONFIRMED_WITHDRAW or CANCELLED.
 - Action: Payee FSP waits for TIPS to update the status as appropriate and will receive a callback on the updated state in TIPS.
- ✓ **FAILURE:** When TIPS does not accept the update confirmation request. Failure can be due to network or bad request.

- Interpretation: The reversal-in update confirmation is not successful. The reversal request state remains INITIATED or RECEIVED.
- Action: The payee FSP should modify request (if need be) and is required to send the update confirmation again.

NOTE: The confirmation process can also be accomplished through the TIPS Portal in case the payee FSP has challenges or issues of sending the update confirmation from their platforms. Refer to the TIPS Portal user manual.

Step 3: Reversal: This is done by payee FSP consuming the TIPS API resource as specified in section 4.3.4. This instructs TIPS to perform the actual reversal transfer. Reversal can ONLY be done when the reversal request is in state CONFIRMED_HOLD or CONFIRMED_WITHDRAW. The outcome, interpretation and actions for this is as follows:

- ✓ SUCCESS: When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The request to reverse is successful.
 - Action: Payee FSP waits for TIPS to process and finalize the reversal and will be notified on the outcome via a callback (step 4).
- ✓ FAILURE: When TIPS does not accept the reversal request. Failure can be due to network or bad request.
 - Interpretation: The reversal request is not successful.
 - Action: The payee FSP should modify request (if need be) and is required to send the reversal request again.

NOTE: The reversal process can also be accomplished through the TIPS Portal in case the payee FSP has challenges or issues of sending the reversal from their platforms. Refer to the TIPS Portal user guide.

Step 4: Reversal callback: This callback notifies the payee FSP on successful completion or errors during performing a transfer reversal.

The payee FSP will receive one of the following two possible callbacks depending on completion status of a transfer.

- i. **Success callback:** to notify successful completion of a reversal process.

This is done by TIPS consuming the payee FSP API resource as specified in section 4.3.5

- Interpretation: The reversal has been successfully processed by TIPS. The TIPS settlement account for original payee FSP is debited and that for original payer FSP is credited. The reversal state changes to REVERSED.
- Action: The payee FSP marks this reversal as completed.

- ii. **Unsuccessful callback:** to notify unsuccessful processing of the reversal.

This is done by TIPS consuming the original payee FSP API resource specified in section 4.3.6

- Interpretation: The reversal is not successful. The reversal message request state remains in CONFIRMED_HOLD or CONFIRMED_WITHDRAW.
- Action: The payee FSP should modify the request based on the error information provided and send the reversal (step 3) again.

Appendix 1.8: Guide to perform reversal messaging enquiry

Reversal messaging enquiry enables a payer or payee FSP to request information of a particular reversal message request. The requesting FSPs must have participated in that transfer and reversal request as either payer or payee, otherwise the enquiry shall be rejected.

Reversal message enquiry is done by an FSP consuming the TIPS API resource as specified in section 4.3.3. The outcome for this can be as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The request is received and processed successfully at TIPS and returns a response.
 - Action: The action to be performed is as follows:

Response & Interpretation	FSP Action
Enquiry returns reversal state (the value of parameter reversalState) equal to [REVERSED]. Means the transfer has been successfully reversed.	Payer and Payee FSP to finalize and complete on their side
Enquiry returns reversal states (the value of parameter reversalState) equals to [CONFIRMED_HOLD, CONFIRMED_WITHDRAW]. Means the payee FSP has successfully hold funds from the payee customer and will reverse upon confirmation or the payee customer has withdrawn the funds.	Payee FSP: If it is hold then confirm and thereafter send reversal request. If the withdrawn, then customer to fund their account if possible and then reverse. Payer FSP: Hold and wait for reversal to be finalized.
Enquiry returns reversal states (the value of parameter reversalState) equals in [INITIATED, RECEIVED]. Means the reversal request is still under processing.	Payer FSP: No action, follow up with the payee FSP to finalize the reversal request. Payee FSP: Should process and finalize the reversal request
Enquiry returns reversal states (the value of parameter reversalState) equals in [CANCELLED]. Means the reversal request has been rejected by payee FSP.	Payer FSP: Can initiate a new reversal request if need be. Payee FSP: No action

- ✓ **FAILURE:** When TIPS does not accept the request. Failure can be due to network or bad request.
 - Interpretation: The reversal message enquiry is not successful.
 - Action: The FSP should verify and modify the request as needed and try again.

Appendix 1.9: Guide to perform reversal transfer enquiry

Reversal transfer enquiry enables a payer or payee FSP to request information of a particular reversed transfer. The requesting FSP must have participated in that transfer as either payer or payee, otherwise the enquiry shall be rejected.

Reversal transfer enquiry is done by an FSP consuming the TIPS API resource as specified in section 4.3.7. The outcome for this can be as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The request is received and processed successfully at TIPS and returns a response.
 - Action:

Response & Interpretation	FSP Action
Enquiry returns reversal state (the value of parameter reversalState) equal to [REVERSED]. Means the transfer has been successfully reversed.	The transfer is reversed and the FSPs should commit.
Enquiry returns error response that “transfer is not found” with error code 3202 as follows: <pre> {"errorInformation": {"errorCode":3202,"errorDescription":"Provided Payer reference or payer FSP ID was not found."} } </pre> Means there is no reversed transfer.	No action.

- ✓ **FAILURE:** When TIPS does not accept the request. Failure can be due to network or bad request.
 - Interpretation: The reversal message enquiry is not successful.
 - Action: The FSP should verify and modify the request as needed and try again.

Appendix 1.10: Guide to perform requestPay-out

The FSP (who will become the payee) initiates a requests for specific payment from a FSP (who will become a payer) so that they receive the payments associated with their request to pay.

The requestPay-out is for the payee FSP only and is accomplished in four steps as follows:

- ✓ Step 1: Initiation
- ✓ Step 2: Update confirmation (when need be)
- ✓ Step 3: Update callback (when need be)
- ✓ Step 4: Transfer

Step 1: Initiation: requestPay-out initiation is done by payee FSP consuming the TIPS API resource as specified in section 4.4.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The requestPay-out initiation is successful. The state of request to pay changes to INITIATED.
 - Action: The payee FSP waits for the request to be processed and will be notified via update callback (step 3) or can decide to update the request to pay (step 2) or wait to receive a transfer-in (step 4).
- ✓ **FAILURE:** When TIPS does not accept the request. The failure can be caused by a number of factors such as network problems, incorrect request, processing errors. TIPS responds to payee FSP with details of failure encountered as specified in the respective error response.
 - Interpretation: The requestPay-out initiation is not successful.
 - Action: The payee FSP should verify and modify the request based on the error message or check network connectivity and try again.

Step 2: Update confirmation: requestPay-out update confirmation is done by payee FSP consuming the TIPS API resource as specified in section 4.4.2. The payee FSP uses the

update confirmation to update details of the request to pay or to CANCEL it so that no more payments are made with respect to that request to pay. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The update confirmation is successful.
 - Action: Payee FSP waits for TIPS to update the status as appropriate and will receive a callback on the updated status (step 3).
- ✓ **FAILURE:** When TIPS does not accept the update confirmation request. Failure can be due to network or bad request.
 - Interpretation: The requestPay-out update confirmation is not successful.
 - Action: The payee FSP should modify request (if need be) and is required to send the update confirmation again.

Step 3: Update callback: This callback notifies the payee FSP on the processing status of their requestPay-out request as updated by the payer FSP. When the payer FSP rejects the request to pay, the payee FSP is notified using this callback. This is done by TIPS consuming the payee FSP API resource as specified in section 4.4.2. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payee FSP accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The update callback is successful.
 - Action: The payee FSP can perform different actions based on the response provided.
- ✓ **FAILURE:** When TIPS fails to deliver the callback or payee FSP fails to accept the callback due to network problems or incorrect request respectively.
 - Interpretation: The payee FSP is not updated on the request to pay status.
 - Action: TIPS will try to re-send the update callback upon request. Also, the payee FSP can perform request to pay enquiry process as guided in Appendix 1.12 and use the response to proceed with further steps.

Step 4: Transfer: requestPay-out transfer process is for payee FSP to receive the payments with respect to the request to pay they initiated previously. The transfer process follows the transfer-in process which is documented in Appendix 1.4

Appendix 1.11: Guide to perform requestPay-in

The payer FSP receives the request to pay and makes payment with respect to a request to pay.

The requestPay-in is for the payer FSP only and is accomplished in four steps as follows:

- ✓ Step 1: Initiation
- ✓ Step 2: Update confirmation (when need be)
- ✓ Step 3: Update callback
- ✓ Step 4: Transfer

Step 1: Initiation: requestPay-in initiation is done by TIPS consuming the payer FSP API resource as specified in section 4.4.1. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payer FSP accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The requestPay-in initiation is successful. The state of request to pay changes to RECEIVED.
 - Action: TIPS waits for the request to be processed by the payer FSP and payer FSP will be required to send update confirmation (step 2) or send the transfer-out (step 4).
- ✓ **FAILURE:** When payer FSP does not accept the request. The failure can be caused by a number of factors such as network problems, incorrect request, processing errors. TIPS responds to payee FSP with details of failure encountered as specified in the respective error response.
 - Interpretation: The requestPay-in initiation is not successful.
 - Action: TIPS will try to re-send the request again or the payer FSP can proceed with enquiry and use the response to proceed.

Step 2: Update confirmation: requestPay-in update confirmation is done by payer FSP consuming the TIPS API resource as specified in section 4.4.2. The payer FSP uses the update confirmation to REJECT the request to pay and it will accept no more payments with respect to that request to pay. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The update confirmation is successful.
 - Action: Payer FSP waits for TIPS to update the status as appropriate and will receive a callback on the updated status (step 3).
- ✓ **FAILURE:** When TIPS does not accept the update confirmation request. Failure can be due to network or bad request.
 - Interpretation: The requestPay-in update confirmation is not successful.
 - Action: The payer FSP should modify request (if need be) and is required to send the update confirmation again.

Step 3: Update callback: This callback notifies the payer FSP on the processing status of their requestPay-in request as updated by the payee FSP. When the payee FSP cancels or updates the request to pay, the payer FSP is notified using this callback. This is done by TIPS consuming the payee FSP API resource as specified in section 4.4.2. The outcome, interpretation and actions for this is as follows:

- ✓ **SUCCESS:** When payer FSP accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The update callback is successful.
 - Action: The payee FSP can perform different actions based on the response provided.
- ✓ **FAILURE:** When TIPS fails to deliver the callback or payer FSP fails to accept the callback due to network problems or incorrect request respectively.
 - Interpretation: The payer FSP is not updated on the request to pay status.

- Action: TIPS will try to re-send the update callback upon request. Also, the payer FSP can perform request to pay enquiry process as guided in Appendix 1.12 and use the response to proceed with further steps.

Step 4: Transfer: requestPay-in transfer process is for payer FSP to send the payments with respect to the request to pay. The transfer process follows the transfer-out process which is documented in Appendix 1.3.

Appendix 1.12: Guide to perform request to pay enquiry

Request to pay enquiry enables a payer or payee FSP to request information of a particular request to pay.

Request to pay enquiry is done by an FSP consuming the TIPS API resource as specified in section 4.4.3. The outcome for this can be as follows:

- ✓ **SUCCESS:** When TIPS accepts the request and responds successfully as specified in the respective success response.
 - Interpretation: The request is received and processed successfully at TIPS and returns a response.
 - Action:

Response & Interpretation	FSP Action
Enquiry returns request pay state (the value of parameter requestPayState) equal to [FULL_PAID]. Means the request to pay has been paid in full. No more payments can be made to that request to pay.	Marks this request to pay as fully paid. No more payments accepted.
Enquiry returns request pay state (the value of parameter requestPayState) equal to [INITIATED, RECEIVED, PARTIAL_PAID]. Means the request to pay is still in processing.	The payer FSP can make more payments to the request to pay.
Enquiry returns request pay state (the value of parameter requestPayState) equal to [REJECTED, CANCELLED]. Means the request to pay has been	The request to pay has reached final state. No more

rejected by payer FSP or cancelled by payee FSP respectively. No more payments can be made to that request to pay	operations are allowed.
---	-------------------------

- ✓ **FAILURE:** When TIPS does not accept the request. Failure can be due to network or bad request.
 - Interpretation: The reversal message enquiry is not successful.
 - Action: The FSP should verify and modify the request as needed and try again.

Appendix 1.13: Guide to perform Funds-In

Adding funds into the participant's TIPS Settlement Account. Upon successful funds-in the participant's TIPS Settlement Account gets a credit and subsequently their TIPS Position increase by the amount equivalent to the funds-in amount.

Funding the TIPS Settlement Account is done through Tanzania Interbank Settlement System (TISS) by sending an MT 202 transaction message. The following should be noted:

- a. You can **ONLY** fund-in the TIPS Settlement Account through TISS
- b. Care is needed when providing information on MT 202 to avoid funding inappropriate participant TIPS Settlement Account
- c. Participants who have no BOT Clearing Accounts can make arrangements with other participants who have BOT Clearing Accounts to sponsor them in funding their TIPS Settlement Account.

The following shall be considered as the possible outcomes of the funds-in process:

- ✓ **SUCCESS:** When an MT 202 transaction message is successfully ACKed by TISS and the funds successfully hits the TIPS Settlement Account and participant's TIPS Position increases. Immediately after the TIPS Position increases, the participant will receive email notification (delivery email addresses must be registered in TIPS) to inform on the success of the process indicating the amount funded-in and the new position after funds-in.

- NOTE: If the transaction is successful in TISS but not reflected in TIPS Position for more than time specified in SLAs, participant should follow up with TIPS SUPPORT.
- ✓ FAILURE: When an MT 202 transaction message is NACKed in TISS. This means the funds-in process is unsuccessful and the participant should verify the information and try again.

The MT 202 funds-in transaction message format is like below:

```
{<messageHeader>} {4:
:20:<reference>
:21:<relatedReference>
:32A:<dateCurrencyAmount>,
:52A:<senderBIC>
:58A:/9927666660
TANZTZX
:72:/REC/TIPS/<orderingCustomerBIC>/S
-}
```

Description of MT 202 funds-in transaction message variables:

Variable	Description and Examples
<messageHeader>	These are message headers automatically generated when an MT 202 funds-in transaction message is composed. Example: {1:F01COPUTZTZAXXX9040544872} {2:I202TANZTZXN} {3:{103:TIS}{121:0d4f31dd-f282-4a22-91fb-2737ca428d0a}}
<reference>	This is the payment reference set by the sender.
<relatedReference>	This is an optional related payment reference set by the sender
<dateCurrencyAmount>	Date currency and amount to fund in. Example: 230128TZS2000000000
<senderBIC>	This is the BIC of the participant whose BOT Clearing Account will be debited.
<orderingCustomerBIC>	This is the BIC of the participant whose TIPS Settlement Account will be credited.

	NOTE: a. If a participant is a non-bank the value of <orderingCustomerBIC> will be a dummy BIC provided by TIPS. b. If a participant is sponsoring another participant (sponsored participant) to fund-in their TIPS Settlement Account, they are required to put the BIC of the sponsored participant in this field.
--	---

Appendix 1.14: Guide to perform Funds-Out

Withdrawing funds from the participant's TIPS Settlement Account. Upon successful funds-out the participant's TIPS Settlement account gets a debit and subsequently their TIPS Position decreases by the amount equivalent to the funds-out amount.

The following should be noted:

- a. The FSP can ONLY fund-out from their own TIPS Settlement Account.
- b. The FSP can ONLY fund-out the TIPS Settlement account through a BOT Clearing Account
- c. FSPs who have no BOT Clearing Accounts (sponsored participants) can make arrangements with other participant to sponsor them in funding out the TIPS Settlement Account. So that, when the withdrawn funds hit the sponsor's BOT Clearing Account are automatically redirected to their account sitting at their sponsor's commercial bank. This account must be configured in TIPS.

Withdrawing funds from the TIPS Settlement Account is done through the TIPS Portal and involves a three level approval process as follows:

LEVEL 1: FSP Initiation: The portal user (for a particular FSP) with required role initiates the process by entering desired amount to fund-out and reference(s). A participant cannot withdraw funds more than their current TIPS Position. Upon success initiation of the funds-out transaction, the amount is reserved so that it cannot be utilized to provision for outgoing transactions and its state changes to RESERVED. Further, notification email and portal alert

are sent to portal users next in-line of the approval process informing them to proceed with LEVEL 2 below.

LEVEL 2: FSP Approval: The portal user (for a particular FSP) with required role reviews the initiated funds-out request and can either APPROVE or REJECT the request.

- ✓ When REJECT: The funds-out process ends here and the reserved funds are released. The state of transaction changes to ABORTED. Further, email notification and portal alert are sent to the initiator of the request to notify rejection of the request.
- ✓ When APPROVE: The funds-out request processing for FSP ends here, notification email and portal alert are sent to portal users next in-line of the approval process informing them to proceed with LEVEL 3.

LEVEL 3: BOT Approval: The portal user (BOT staff) with the required role reviews the funds-out request and can either APPROVE or REJECT the request.

- ✓ When REJECT: The funds-out process ends here and the reserved funds are released. The state of transaction changes to ABORTED. Further, email notification and portal alert are sent to the initiator of the request to notify rejection of the request.
- ✓ When APPROVE: TIPS sends instructions to TISS to debit participant's TIPS Settlement Account and credit respective BOT Clearing Account. For sponsored participants, funds shall further be redirected to their Trust Accounts sitting at the sponsor's commercial bank. The transaction state changes to CONFIRMED and TIPS waits for the feedback from TISS in order to finalize the funds-out process. Further, email notification and portal alert are sent to the initiator to notify approval of the request.
 - When the feedback is received from TISS, TIPS finalizes the funds-out transaction by reducing the participants TIPS Position and transaction state changes to COMMITTED. Immediately after the TIPS Position decreases, the participant will receive email notification (delivery email addresses must be registered in TIPS) to inform on the success of the process indicating the amount funded-out and the new position after funds-out.

- NOTE: If the funds-out transaction remains CONFIRMED for more than time specified in SLAs, implies there is delayed feedback from TISS and participant should follow up with TIPS SUPPORT.

Appendix 1.15: Guide to perform Reconciliation

Reconciliation in TIPS is done to ensure all operations performed in FSP's systems matches those that recorded in TIPS. It is important to note that, each FSP **MUST** consider TIPS as a central point of truth and reconcile against TIPS and NOT against peer FSPs.

TIPS is flexible to operate multiple clearing sessions within a day as need be. This provides flexibility for FSPs to perform intra-day reconciliations and be able to resolve customer disputes timely.

It is important to note that, all transactions in TIPS are irrevocable. That is, once a transaction is finalized it can NOT be altered.

When performing reconciliation, the FSP shall consider the following:

- a. Are all the operations (transfers, payments, reversals, funds-in or funds-out) recorded in FSP's systems also recorded in TIPS? If not recorded in TIPS, then they should not be in FSPs system (or marked as NOT SUCCESSFUL in FSP system).
- b. For those that are available in TIPS, what is the state of these operations? The following shall apply for different states as follows:

TIPS transaction State	Reconciliation Action
COMMITTED	For those that are COMMITTED means they have they affected FSP's TIPS Settlement Account. The FSP proceeds with reconciliation of the TIPS Settlement Account. Note: ONLY successfully COMMITTED transactions are the ones that affect the participant's TIPS Settlement Account.
ABORTED or INVALID	The FSP marks these transactions as NOT successful.

	Note: ABORTED or INVALID transactions do not affect the FSP's TIPS Settlement Account.
All other states: RECEIVED, VALIDATED, RESERVED, ACKNOWLEDGED, CONFIRMED	The FSP marks these transactions as pending and will reconcile in the next cycle of reconciliation when they have been completed. These transactions have not affect the TIPS Settlement Account.

Below are the different types of information TIPS provides to FSPs to assist in reconciliation.

a. Reconciliation files

These are CSV files containing all FSP's operations performed within a certain clearing session/window. TIPS prepares and provides two sets of files, one for successful transactions (COMMITTED) and the other one for unsuccessful (INVALID, ABORTED and all intermediate states). See details and formats of the files in section 4.5.2 and 4.5.3. At the close of every clearing window, these two sets of files are made available on the TIPS FTP servers for FSPs to download. Every FSP will be provided with their own credentials to access and download the files.

b. Enquiries through TIPS Portal

The FSPs can login the TIPS Portal to view transactions and their current state at any time they wish to do so. They have options to enquire for individual transaction or bulk of transactions and can export or download the desired transactions to Excel.

c. Settlement notification API

TIPS sends settlement notification to all FSPs at the close of the clearing session. This notification contains information on the volume and value of transactions (incoming and outgoing) performed during the session as well as balances of the FSP's Accounts as at the close of the clearing session. TIPS sends this notification by consuming the FSPs API resource specified in section 4.5.1

7. Appendix 2: Supported Use Cases

TIPS API supports a number of use cases for different scenarios as described in the table below.

No.	Use Case	Sub Use Case	Scenario
1	P2P – Person to Person	W2W, W2B, B2B, B2W	TRANSFER
2	P2B – Person to Business	W2W, W2B, B2B, B2W	TRANSFER
3	P2B with request to pay	W2W, W2B, B2B, B2W	PAYMENT_REQUESTPAY
4	B2B – Business to Business	W2W, W2B, B2B, B2W	TRANSFER
5	B2P – Business to Person	W2W, W2B, B2B, B2W	TRANSFER
6	P2M – Person to Merchant	W2W, W2B, B2B, B2W	PAYMENT
7	P2M with request to pay	W2W, W2B, B2B, B2W	PAYMENT_REQUESTPAY
8	B2M – Business to Merchant	W2W, W2B, B2B, B2W	PAYMENT
9	P2G – Person to Government	W2B, B2B	PAYMENT
10	B2G – Business to Government	W2B, B2B	PAYMENT
11	P2P – Person to Person	W2W, W2B, B2B, B2W	TRANSFER_IMT
12	P2B – Person to Business	W2W, W2B, B2B, B2W	TRANSFER_IMT
13	B2B – Business to Business	W2W, W2B, B2B, B2W	TRANSFER_IMT
14	B2P – Business to Person	W2W, W2B, B2B, B2W	TRANSFER_IMT

Key: W – Wallet, B – Bank, W2B – Wallet to Bank

Appendix 2.1: Merchant Payments

These are payments to merchants through TIPS using the QR (TANQR) code or a merchant code (commonly known as “Lipa Namba”). This use case can also be used for payment of a specific bill identified by a certain bill reference. For more information about the TANQR please refer to TANQR CODE STANDARDS.

The merchant payments in TIPS can be initiated using two methods.

- i. Scanning the QR (TANQR) code to extract information
- ii. Using the merchant code displayed on the merchant QR (TANQR) sticker

Scanning the QR (TANQR) code to extract information

Payer FSP through their customer's mobile application will capture the following information upon successful scanning of the merchant code:

Information	Obtained From	Example/Value
FSP ID	Field 26 – sub id 01 (3 postfix digit numbers)	e.g. 01003 or 02501. Then the FSP ID is 003 or 501
Payee Category ID	Field 26 – sub id 01 (2 prefix digit numbers)	e.g. 01003 or 02501. Then the ID is 01 or 02
Merchant code	Field 26 – sub id 02	e.g. 00123456
Transaction Currency	Field 53	i.e. TZS
Merchant Name	Field 59	e.g. ABC Store
Postcode	Field 61	e.g. 11000
This is the full QR string	Field 62	e.g. 6221030800112349070511002
Verify Cyclic Redundancy Check	Field 63	e.g. C5DF
Transaction amount	Field 54 of the dynamic QR code or entered by customer	e.g. 5000.00
billRef	Field 62 sub id 05 of the dynamic QR code or entered by customer	e.g. ABC345678QW3456

Using the merchant code displayed on the Merchant QR (TANQR) sticker

Payer FSP through their customer's mobile application or USSD will capture the following information entered by a customer:

Information	Obtained From	Example/Value
Merchant code (“Lipa Namba”)	Displayed on the code	e.g. 00123456
Transaction amount	Entered by customer	
billRef	Specific bill reference entered by customer used for paying a bill	e.g. ABC345678QW3456

Lookup process

After obtaining the information of the merchant above, the payer FSP generates more information (that are not captured from code or customer input) before sending lookup request to TIPS. The information is as below:

Information	Obtained From	Example/Value
Transaction Date and Time	Generated by a payer system	Current date and time
identifierType	Generated by payer system	BUSINESS

Thereafter, the payer FSP proceeds with the lookup and TIPS responds as specified in section 4.1.1

NOTE: The lookup response will NOT contain additionalInfo object.

Transfer process

Upon successful lookup response, the payer FSP generates more information (that are not part of the lookup process) and thereafter proceeds with transfer process as specified in section 4.2.1

Information	Obtained From	Example/Value
payerRef	Generated by payer system	
payer	Generated by payer system	Information of the payer
Transaction Date and Time	Generated by a payer system	
scenario	Generated by payer system	PAYMENT

The formulation of the transfer request to TIPS follows the flow and specifications in section 4.2.1 with exception of additional information as detailed in the request below.

✓ Request:

```
POST /transfers HTTP/1.1

Accept: application/json
Channel: "IB"
Content-Type: application/json
Date: 2020-10-14 10:46:05
Fsp-source: "003"
Fsp-destination: "004"
Lookup-Id: 9dcb27b1-90b1-4547-b7fb-beac477bf2ab
```

```

{
  "payerRef": "003-7k084ehse56",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "003",
    "fullName": "Ally Mtoni",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    },
  },
  "payee": {
    "identifierType": "BUSINESS",
    "identifier": "00123456",
    "fspId": "004",
    "fullName": "ABC Store",
    "accountCategory": "MERCHANT",
    "accountType": "BANK",
    "identity": {
      "type": "TIN",
      "value": "123123123"
    },
    "additionalInfo": {
      "field62": "6221030800112349070511002",
      "billRef": "ABC345678QW3456",
    },
  },
  "amount": {
    "amount": "100",
    "currency": "TZS"
  },
  "endUserFee": {
    "amount": "5",
    "currency": "TZS"
  },
  "transactionType": {
    "scenario": "PAYMENT",
    "initiator": "PAYER",
    "initiatorType": "CONSUMER"
  },
  "description": "Payment for goods",
}

```

✓ Response:

```

HTTP/1.1 202 Accepted

{
  "payerRef": "003-7k084ehse56",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

```

✓ Description of variables:

Note: ONLY additional/key variables for this use case are described here. For a comprehensive description of all variables, refer to the specific API in section 4 above.

Variable	Description and Examples
payee	accountCategory: This must be MERCHANT
additionalInfo	- field62: This is the full QR code string. This is optional - billRef: Specific bill reference entered by customer. This is optional
transactionType	scenario: This must be PAYMENT

Finally, after the payment completes in TIPS, it sends the final notification/callback to both the payer and payee FSP with all details as below:

✓ Request:

```
PUT /transfers/003-7k084ehse56 HTTP/1.1

Accept: application/json
Content-type: application/json
Date: 2020-10-14 10:46:13
Fsp-source: "TIPS"
Fsp-destination: "003"
{
  "payeeRef": "004-xyz123hy999",
  "switchRef": "021950bd-7e98-4b67-adb9-321eb0bad792",
  "completedTimestamp": "2020-10-14 10:46:13",
  "transferState": "COMMITTED",
  "settlementWindowId": "115",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "255556789922",
    "fspId": "003",
    "fullName": "Ally Mtoni",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "19831007175010005135"
    },
  },
  "payee": {
    "identifierType": "BUSINESS",
    "identifier": "00123456",
    "fspId": "004",
    "fullName": "ABC Store",
    "accountCategory": "MERCHANT",
    "accountType": "BANK",
    "identity": {
      "type": "TIN",
      "value": "123123123"
    },
  },
  "additionalInfo": {
```

```

"field62": "6221030800112349070511002",
"billRef": "ABC345678QW3456",
}

```

✓ Response:

HTTP/1.1 202 Accepted

```

{
  "payerRef": "003-7k084ehse56",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

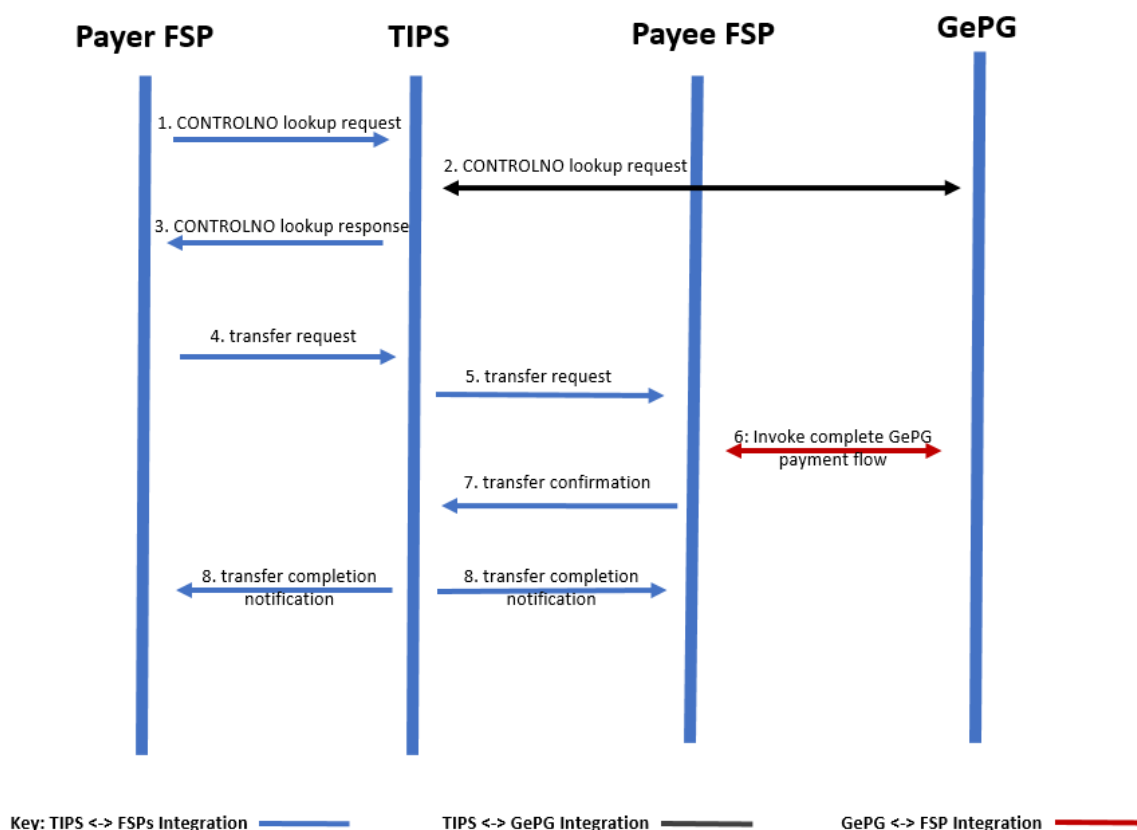
```

Appendix 2.2: Government payments

This use case supports payments to the government using a GePG control number. The FSP can do lookup and thereafter make a payment to a particular GePG control number. This covers the P2G and B2G payment use cases.

The process and flow for the government payments in TIPS is as shown in the diagram below:

Government Payments Flow



Lookup process

To perform lookup, the payer FSP shall submit to TIPS the control number as well as the currency and amount to be paid. The formulation of lookup request and response is as detailed below:

NOTE: amount and currency are mandatory when sending a lookup request to TIPS.

✓ Request:

```
GET /parties/CONTROLNO/995350001916/sync?amount=700000&currency=TZS

Accept: application/json
Content-Type: application/json
Date: 2020-10-14 10:45:44
Fsp-source: "002"
Fsp-destination: " "
```

✓ Response:

```
HTTP/1.1 200 OK

{
  "identifierType": "CONTROLNO",
  "identifier": "995350001916",
  "fspId": "004",
  "fullName": "Muhimbili National Hospital",
  "accountCategory": "GOVERNMENT",
  "accountType": "BANK",
  "identity": {
    "type": "",
    "value": ""
  },
  "additionalInfo": {
    "billHdr": {
      "lookUpId": "de50d310-db44-4964-b2c3-799bade1b430",
      "custCtrNum": "995350001916",
      "tendrAmt": "700000",
      "totalPayAmt": "700000",
      "payType": "2",
      "entryCnt": "1",
      "billStsCode": "7101",
      "billStsDesc": "Successful",
      "payerPspCode": "PSP868"
    },
    "billDtls": [
      {
        "billDtl": {
          "spCode": "SP99535",
          "spName": "Muhimbili National Hospital",
          "billCtrNum": "995350001916",
          "billAmt": "700000.00",
          "minPayAmt": "700000.00",
          "maxPayAmt": "700000.00",
          "payAmt": "700000",
          "ccy": "TZS",
          "billExprDt": "2024-08-27T23:59:59",
          "billDesc": "Malipo ya Kibanda",
          "custName": "Simon Musa",
          "custCellNum": "255758154019",
          "custEmail": "smusa@bot.go.tz",
          "accDtls": {
            "accDtl": [
              {
                "pspCode": "PSP869",
```

```

        "fspId": "900",
        "bankBic": "DASUTZT0"
      },
      {
        "pspCode": "PSP866",
        "fspId": "901",
        "bankBic": "TAINTZT0"
      },
      {
        "pspCode": "PSP868",
        "fspId": "004",
        "bankBic": "NMIBTZT0"
      }
    ],
    "rsv1": {},
    "rsv2": {},
    "rsv3": {}
  }
}

```

✓ **Description of lookup variables:**

Note: ONLY additional/key variables for this use case are described here. For a comprehensive description of all variables, refer to the specific API in section 4 above.

Variable	Description and Examples
additionalInfo	- billHdr: ✓ lookupId – the lookup ID returned by TIPS during lookup ✓ custCtrNum – the control number ✓ tenderAmt – amount the customer will pay (in two decimal places) ✓ totalPayAmt – total payable amount i.e. total amount eligible for bill(s) with two (2) decimal places ✓ payType – 1 for Pay All Bill Control Numbers at once, 2 for Pay any Bill Control ✓ entryCnt – total Number of bill entries contained in the details payload ✓ billStsCode – bill status code ✓ billStsDesc – bill status description ✓ payerPspCode – thePSP code of the payer FSP. - billDtls: ✓ spCode – service provider code ✓ spName – service provider name ✓ billCtrNum – bill control number to be used during crediting respective service provider collection account,

	included in bank statement and included in transactional reconciliation submission ✓ billAmt – billed Amount with two (2) decimal places ✓ minPayAmt – minimum payable amount for billed amount of the respective bill with two (2) decimal places ✓ maxPayAmt – maximum payable amount for billed amount of the respective bill with two (2) decimal places ✓ payAmt – payable amount i.e amount eligible for respective bill with two (2) decimal places ✓ ccy – currency ✓ billExprDt – bill expiry date ✓ billDesc – bill description ✓ custName – customer name ✓ custCellNum – customer mobile or cell number ✓ custEmail – customer email ✓ accDtls – details of eligible payee banks. This will include their pspCode, fspId and bankBIC • rsv1: reserved field • rsv2: reserved field • rsv3: reserved field
--	--

Transfer process

Upon successful lookup response, the payer FSP selects one of the payee FSPs (from the possible payee FSPs returned during lookup) and proceeds with making a payment by sending the transfer to TIPS as specified below:

✓ Request:

```
POST /transfers HTTP/1.1

Accept: application/json
Channel: "IB"
Content-Type: application/json
Date:2020-10-14 10:46:05
Fsp-source: "002"
Fsp-destination: "004"
Lookup-Id:"de50d310-db44-4964-b2c3-799bade1b430"

{
  "payerRef": "002-186690a5-0e00-4e97-8357-ad9cdc891d73",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "0712347089",
    "fspId": "002",
    "fullName": "ASHA MAZIKU",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "5678912345678912"
    }
  }
}
```

```

    },
    "additionalInfo": {
      "payerCellNum": "0748432212900",
      "payerPspCode": "PSP868",
      "payerEmail": "amaziku@bot.go.tz",
      "payerReceipt": "DFTO45678RFG"
    }
  },
  "payee": {
    "identifierType": "CONTROLNO",
    "identifier": "995350001916",
    "fspId": "004",
    "fullName": "Muhimbili National Hospital",
    "accountCategory": "GOVERNMENT",
    "accountType": "BANK",
    "identity": {
      "type": "",
      "value": ""
    }
  },
  "amount": {
    "amount": "700000",
    "currency": "TZS"
  },
  "endUserFee": {
    "amount": "100",
    "currency": "TZS"
  },
  "transactionType": {
    "scenario": "PAYMENT",
    "initiator": "PAYER",
    "initiatorType": "SENDER"
  },
  "description": "Test GePG Payment",
  "additionalInfo": {
    "spCode": "SP99535"
  }
}

```

✓ Response:

```

HTTP/1.1 202 Accepted

{
  "payerRef": "002-186690a5-0e00-4e97-8357-ad9cdc891d73",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

```

Thereafter, the payee FSP after receiving the payment request proceeds with validating it with GePG where they will obtain additional references from GePG and respond with a callback to TIPS as detailed below:

✓ Request:

```

PUT /transfers/002-186690a5-0e00-4e97-8357-ad9cdc891d73 HTTP/1.1

Accept: application/json
Content-type: application/json
Date: 2020-10-14 10:46:10
Fsp-source: "004"
Fsp-destination: "003"

{
  "payeeRef": "004-6a92b8-7010-4499-aa3e-4b6f8f450f",
  "transferState": "COMMITTED",

```

```

"reasonCode": "60",
  "additionalInfo": {
    "payRefId": "CSC-WRF3-9BPW",
    "transactionId": "0237-vnz-gwr-6zb7",
    "collectionAccountNumber": "05A21G24908800"
  }
}

```

✓ Response:

HTTP/1.1 202 Accepted

```

{
  "payerRef": "002-186690a5-0e00-4e97-8357-ad9cdc891d73",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

```

Finally, after the payment completes in TIPS, it sends the final notification/callback to both the payer and payee FSP with all details as below:

✓ Request:

PUT /transfers/002-186690a5-0e00-4e97-8357-ad9cdc891d73 HTTP/1.1

Accept: application/json
 Content-Type: application/json
 Date: 2024-05-31 08:38:22
 fsp-source: switch
 fsp-destination: 002

```

{
  "payeeRef": "004-6a92b8-7010-4499-aa3e-4b6f8f450f",
  "switchRef": "77af4556-1155-47ac-a210-be96f081a569",
  "completedTimestamp": "2024-05-31 08:40:22.987",
  "settlementWindowId": "244",
  "transferState": "COMMITTED",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "0712347089",
    "fspId": "002",
    "fullName": "ASHA MAZIKU",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {"type": "NIN",
      "value": "5678912345678912"},
    "additionalInfo": {
      "payerCellNum": "048432212900",
      "payerPspCode": "PSP868",
      "payerEmail": "amaziku@bot.go.tz",
      "payerReceipt": "DFTO45678RFG"
    }
  },
  "payee": {
    "identifierType": "CONTROLNO",
    "identifier": "995350001916",
    "fspId": "004",
    "fullName": "Muhimbili National Hospital",
    "accountCategory": "GOVERNMENT",
    "accountType": "BANK",
    "identity": {"type": "",
      "value": ""}
  },
  "additionalInfo": {
    "spCode": "SP99535",
    "usdPayChnl": "IB",
    "payeePspCode": "800",
  }
}

```

```

    "payRefId": "CSC-WRF3-9BPW",
    "transactionId": "0237-vnz-gwr-6zb7",
    "collectionAccountNumber": "05A21G24908800"
  }
}

```

✓ Response:

```

HTTP/1.1 202 Accepted

{
  "payerRef": "002-186690a5-0e00-4e97-8357-ad9cdc891d73",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

```

To enquire and obtain the status and details of a particular government payment to TIPS, the payer or payee FSP shall enquire as detailed below:

✓ Request:

```

GET /transfers/002-186690a5-0e00-4e97-8357-ad9cdc891d73 HTTP/1.1

Accept: application/json
Content-Type: application/json
Date: 2024-05-30 11:40:22
fsp-source: 002
fsp-destination: 001

```

✓ Response:

```

HTTP/1.1 200 OK

{
  "payerRef": "002-186690a5-0e00-4e97-8357-ad9cdc891d73",
  "payer": {
    "identifierType": "MSISDN",
    "identifier": "0712347089",
    "fspId": "002",
    "fullName": "ASHA MAZIKU",
    "accountCategory": "PERSON",
    "accountType": "WALLET",
    "identity": {
      "type": "NIN",
      "value": "5678912345678912"
    },
    "additionalInfo": {
      "payerCellNum": "048432212900",
      "payerPspCode": "PSP868",
      "payerEmail": "amaziku@bot.go.tz"
    }
  },
  "payee": {
    "identifierType": "CONTROLNO",
    "identifier": "995350001916",
    "fspId": "004",
    "fullName": "Muhimbili National Hospital",
    "accountCategory": "GOVERNMENT",
    "accountType": "BANK",
    "identity": {
      "type": "",
      "value": ""
    }
  },
  "amount": {
    "amount": "700000",

```

```

    "currency": "TZS"
  },
  "transactionType": {
    "scenario": "PAYMENT",
    "initiator": "PAYER",
    "initiatorType": "SENDER"
  },
  "lookup-id": "165bfa34-297f-44e2-8426-c67e58812f45",
  "transferState": "COMMITTED",
  "transferStateReason": "The transaction is succesfully completed and committed",
  "payeeRef": "004-6a92b8-7010-4499-aa3e-4b6f8f451f",
  "switchRef": "77af4556-1155-47ac-a210-be96f081a569",
  "transferDate": "2024-05-31 08:35:36",
  "completedDate": "2024-05-31 08:40:22",
  "description": "Test GePG Payment",
  "transactionCategory": {
    "useCase": "PERSON-GOVERNMENT",
    "subUseCase": "WALLET-BANK",
    "description": "P2G Simple Wallet to Bank"
  },
  "settlementWindow": {
    "id": 244,
    "description": "New settlement window created at 2024-03-17 12:48:09"
  },
  "additionalInfo": {
    "spCode": "SP99535",
    "usdPayChnl": "IB",
    "payeePspCode": "800",
    "payRefId": "CSC-WRF3-9BPW",
    "transactionId": "0237-vnz-gwr-6zb7",
    "collectionAccountNumber": "05A21G24908800"
  }
}

```

✓ Description of transfer variables:

Note: ONLY additional/key variables for this use case are described here. For a comprehensive description of all variables, refer to the specific API in section 4 above.

Variable	Description and Examples
Lookup-Id	The lookup ID returned by TIPS during lookup process. This is mandatory when sending government payments.
payerCellNum	Customer mobile or cell number
payerPspCode	The PSP code of the Payer FSP. This is given by GePG
payeePspCode	The PSP code of the payee FSP. This is given by GePG
payerEmail	Customer email
payerReceipt	This is the receipt number given/displayed to the payer customers by their respective payer FSPs.
spCode	The service provider code given to each service provider by GePG

payRefId	Qualified GePG Receipt number for the respective transaction. It is the value of PayRefId obtained during bill verification done by payee FSP to GePG
transactionId	Payee FSP Receipt Unique for each transaction, saved in Core banking, included in bank statements, included in transaction reconciliation submission and Issued as customer receipt.
collectionAccountNumber	Credited account number
usdPayChnl	The channel used by payer customer to make payment

Appendix 2.3: International Money Remittances

This use case supports international money remittances. It is used by an FSP to remit money coming to Tanzania (incoming) only.

Lookup process

No changes to the lookup process. This will be as documented in section 4.1.1

Transfer process

Upon successful lookup response, the payer FSP proceeds with transfer process as specified in section 4.2.1. The formulation of the transfer request will be as follows:

✓ Request:

```
POST /transfers HTTP/1.1

Accept: application/json
Channel: "FT"
Content-Type: application/json
Date:2020-10-14 10:46:05
Fsp-source: "002"
Fsp-destination: "004"
Lookup-Id:"de50d310-db44-4964-b2c3-799bade1b430"
{
  "payerRef": "002-186690a5-0e00-4e97-8357-ad9cdc891d73",
  "payer": {
    "identifierType": "BANK",
    "identifier": "548213654892",
    "fspId": "002",
    "fullName": "REMITTANCE ACCOUNT",
    "accountCategory": "BUSINESS",
    "accountType": "BANK",
    "identity": {
      "type": "",
      "value": ""
    },
  },
  "additionalInfo": {
    "orderingCustomerFullName": "Jeremy Jones",
    "orderingCustomerAddress": "115 Michigan",
```

```

      "orderingCustomerNationality": "US",
      "orderingCustomerReason": "Sending money to TZ for shopping",
      "originalAmount": "200.00",
      "originalCurrency": "USD",
      "sendingIntermediary": "PAYHUB"
    },
    "payee": {
      "identifierType": "BANK",
      "identifier": "01258796325 ",
      "fspId": "004",
      "fullName": "Mahija Joseph",
      "accountCategory": "PERSON",
      "accountType": "BANK",
      "identity": {
        "type": "",
        "value": ""
      }
    },
    "amount": {
      "amount": "700000",
      "currency": "TZS"
    },
    "endUserFee": {
      "amount": "100",
      "currency": "TZS"
    },
    "transactionType": {
      "scenario": "TRANSFER_IMT",
      "initiator": "PAYER",
      "initiatorType": "SENDER"
    },
    "description": "Remittance to TZ",
  }
}

```

✓ Response:

HTTP/1.1 202 Accepted

```

{
  "payerRef": "002-186690a5-0e00-4e97-8357-ad9cdc891d73",
  "status": "RECEIVED",
  "datetime": "2020-10-14 10:46:06"
}

```

✓ Description of variables:

Note: ONLY additional/key variables for this use case are described here. For a comprehensive description of all variables, refer to the specific API in section 4 above.

Variable	Description and Examples
additionalInfo	- orderingCustomerFullName: name of the original/first sender - orderingCustomerAddress: address of the original sender - orderingCustomerNationality: nationality of the original sender - orderingCustomerReason: reasons for sending funds - originalAmount: original amount ordered by the original customer - originalCurrency: original currency ordered by the original customer

	• sendingIntermediary: the name of intermediary switch or hub that delivered the money
transactionType	scenario: Must be TRANSFER_IMT

NOTE: Other steps or procedure of the transfer remains the same.