**OUTWARD SERVICES**

**1. GIP Outbound - NEC**

**2. GIP Outbound - FTC**

**\*\*\* API authentication to be added will be basic auth (I will share Username and Password) to use**

**1. GIP Outbound - NEC**

======= 1. IIB Develop REST API for Channels to call =============

API URL: - ***https://10.180.XXX.XX:XXXX/giptransfer***

Expected Sample JSON

============ 1. Request – Name Enquiry =============

{

"operation": "nameenquiry",

    "destBankCode ": "300315",

    "channelRef": "101900166252",

    " accountToDebit": "1090099199418",

    "nameToDebit": "JOE ANOKYE",

    " accountToCredit": "1020820171412",

    " NameToCredit": "OLAM PURCHASE ACCOUNT",

}

====== 2. Mapping of Channel Request to GhIPSS SOAP Request=====

Request msg 🡺Mapping Channel Request to GhIPSS SOAP Request

|  |  |  |
| --- | --- | --- |
| JSON body | SOAP body | Default Values |
| No mapping | Amount | 000000000000 |
| No mapping | datetime | Use format: *YYMMDDHHMMSS*  230630181136 |
| No mapping | TrackingNum | 6 digits unique values |
| No mapping | FunctionCode | 230 |
| No mapping | OrigineBank | 300323 |
| destBankCode | DestBank |  |
| channelRef | SessionID |  |
| No mapping | ChannelCode | 100 |
| accountToDebit | AccountToDebit |  |
| nameToDebit | NameToDebit |  |
| accountToCredit | AccountToCredit |  |
| nameToCredit | NameToCredit |  |
| No mapping | Narration | GIP Transfer |
| No mapping | Login | 300323 |
| No mapping | Password | as shared with sample |

=========== 4. IIB calls to GhIPSS API ==========

GhIPSS URL: ***https://10.180.143.22:7091/GhipssOutgoing\_IIB***

**Request:-**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:com="com.ghipss.gip">

<soapenv:Header>

<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" soapenv:mustUnderstand="1">

<wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="UsernameToken-1">

<wsse:Username>esbtestuser</wsse:Username>

<wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">fidelity,123</wsse:Password>

<wsse:Nonce EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">PCamrOP55EQorWOZ8zOBHA==</wsse:Nonce>

<wsu:Created>2020-10-28T02:45:11.593Z</wsu:Created>

</wsse:UsernameToken>

</wsse:Security>

</soapenv:Header>

<soapenv:Body>

<com:GIPTransactionOp>

<ReqGIPTransaction>

<Amount>000000000000</Amount>

<datetime>230630181136</datetime>

<TrackingNum>125278</TrackingNum>

<FunctionCode>200</FunctionCode>

<OrigineBank>300323</OrigineBank>

<DestBank>300315</DestBank>

<SessionID>101900166252</SessionID>

<ChannelCode>100</ChannelCode>

<AccountToDebit>1090033199418</AccountToDebit>

<NameToDebit>AUGSUTINE JESSON</NameToDebit>

<AccountToCredit>1020820171412</AccountToCredit>

<NameToCredit>JOHN MENSAH</NameToCredit>

<Narration>TEST</Narration>

<Login>300323</Login>

<Password>c941a4ec7bd7937a581e42f29d09cdda2b2ece8459db0b07444c20b9c1e6ac9a538275698af192af5634ca3fdae3dd50160dcc14d6f10c754041d7b74f9a84d2</Password>

</ReqGIPTransaction>

</com:GIPTransactionOp>

</soapenv:Body>

</soapenv:Envelope>

**Response:-**

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns2:GIPTransactionOpResponse xmlns:ns2="com.ghipss.gip">

<GIPTransactionResponse>

<Amount>000000000000</Amount>

<datetime>2306301811</datetime>

<TrackingNum>125278</TrackingNum>

<FunctionCode>230</FunctionCode>

<OrigineBank>300323</OrigineBank>

<DestBank>300315</DestBank>

<SessionID>101900166252</SessionID>

<NameToDebit>AUGSUTINE JESSON</NameToDebit>

<AccountToDebit>1090099199418</AccountToDebit>

<NameToCredit>OLAM PURCHASE ACCOUNT</NameToCredit>

<AccountToCredit>1020820171412</AccountToCredit>

<Narration>TEST</Narration>

<OneTimePassword xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"></OneTimePassword>

<Login xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"></Login>

<Password xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"></Password>

<ActCode>000</ActCode>

<AprvCode>577294</AprvCode>

</GIPTransactionResponse>

</ns2:GIPTransactionOpResponse>

</soap:Body>

=========== 5. Mapping of GhIPSS SOAP response to REST for Channels ==========

Response msg 🡺Mapping SOAP to REST

|  |  |  |
| --- | --- | --- |
| SOAP body | JSON body | values |
| Amount | No mapping |  |
| datetime | No mapping |  |
| TrackingNum | No mapping |  |
| FunctionCode | No mapping |  |
| OrigineBank | No mapping |  |
| DestBank | No mapping |  |
| SessionID | channelRef | As returns by GhIPSS |
| ChannelCode | No mapping |  |
| AccountToDebit | No mapping |  |
| NameToDebit | No mapping |  |
| AccountToCredit | accountToCredit | As returns by GhIPSS |
| NameToCredit | nameToCredit | As returns by GhIPSS |
| Narration | No mapping |  |
| Login | No mapping |  |
| Password | No mapping |  |
| ActCode | statusCode | As returns by GhIPSS |
| AprvCode | aprvCode | As returns by GhIPSS |

=========== 6. IIB response to Channel ==========

{

    "channelRef": "101900166252",

    " accountToCredit": "1020820171412",

    " nameToCredit": "OLAM PURCHASE ACCOUNT",

    " statusCode ": "000"

“aprvCode”: “577294”

}

***\*\*\* Error handling:***

If StatusCode = ‘114’ respond to Channel,

{

“transactionStatus”: “Invalid Credit Account”

}

If StatusCode = ‘119’ respond to Channel with “**Merchant Account not allowed”**

{

“transactionStatus”: “Merchant Account not allowed”

}

Else respond with Channel with

{

“transactionStatus”: “timeout, please try again”

}

**2. GIP Outbound - FTC**

======= 1. IIB Develop REST API for Channels to call=============

API URL:- ***https://10.180.XXX.XX:XXXX/giptransfer***

Expected Sample JSON

============ 1. Request -- FTC =============

{

"operation": "fundstransfer",

"amount": "50",

    "destBankCode ": "300315",

    "channelRef": "101900166252",

    " accountToDebit": "1090099199418",

    "nameToDebit": "JOE ANOKYE",

    " accountToCredit": "1020820171412",

    " nameToCredit": "OLAM PURCHASE ACCOUNT",

"transactionDetail": "GIP PAYMENT FOR GOODS"

}

========= 2. IIB calls CBS CustInfoService to fetch branch code using accountToDebit ============

1. Call CBS REST API with URL: <https://10.180.143.22:7090/CustAccountInfo_JSON>

**Request:-**

{

"operation":"limiteddetail",

"accountNo":"1400161298613"

}

**Response:-**

{

"custAccNo": "1400161298613",

"customerName1": "NAOMI ADUTWUMWAA KYEI",

"branchCode": "012"

}

\*\*\* If CBS return empty tags, it means account number is wrong, so response to Channel with **“Invalid Debit account number”**

**If CBS returns empty tags:**

It will return no value for account number in custAccNo tag and no value for name in customerName1 tag.

Thus:

"custAccNo": "",

"customerName1": ""

=========== 3. Mapping details in Channel Request to invoke CBS RT Call ==========

Request msg 🡺Mapping Channel REST to CBS SOAP

|  |  |
| --- | --- |
| Channel - JSON body | CBA - SOAP body |
| amount | TXNAMT |
| destBankCode | CHQNO |
| channelRef | XREF |
| accountToDebit | TXNACC |
| nameToDebit | No mapping |
| accountToCredit | BENFADDR1 |
| nameToCredit | BENFNAME |
| transactionDetails | NARRATIVE |

==== 4. IIB use branch code, and details in Channel Request to invoke CBS RT Call ====

CBS URL: <https://10.180.142.218:7081/FCUBSRTServiceV1.5>

Operation: CreateTransaction

**Request:**

<?xml version="1.0" encoding="UTF-8"?>

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ret="http://fcubs.ofss.com/service/FCUBSRTService">

<soapenv:Header>

<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" soapenv:mustUnderstand="1">

<wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="UsernameToken-1">

<wsse:Username>esbtestuser</wsse:Username>

<wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">fidelity,123</wsse:Password>

<wsse:Nonce EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">PCamrOP55EQorWOZ8zOBHA==</wsse:Nonce>

<wsu:Created>2015-10-28T02:45:11.593Z</wsu:Created>

</wsse:UsernameToken>

</wsse:Security>

</soapenv:Header>

<soapenv:Body>

<ret:CREATETRANSACTION\_FSFS\_REQ>

<ret:FCUBS\_HEADER>

<ret:SOURCE>GIP</ret:SOURCE> ==>fix

<ret:UBSCOMP>FCUBS</ret:UBSCOMP> ==>fix

<ret:MSGID>00321010103</ret:MSGID> ==> auto-generated(16 max chanracters)

<ret:CORRELID>00430011003</ret:CORRELID> ==> auto-generated(16 max chanracters)

<ret:USERID>GIPUSER</ret:USERID> ==>fix

<ret:BRANCH>701</ret:BRANCH> ==> make a call to CustInfoService using Account number

<ret:MODULEID>RT</ret:MODULEID> ==>fix

<ret:SERVICE>FCUBSRTService</ret:SERVICE> ==>fix

<ret:OPERATION>CreateTransaction</ret:OPERATION> ==>fix

<ret:SOURCE\_OPERATION>CreateTransaction</ret:SOURCE\_OPERATION> ==>fix

<ret:SOURCE\_USERID>GIPUSER</ret:SOURCE\_USERID> ==>fix

</ret:FCUBS\_HEADER>

<ret:FCUBS\_BODY>

<ret:Transaction-Details>

<ret:XREF>101900066199</ret:XREF> ==> fetch from "ChannelRef" in json request

<ret:PRD>MS38</ret:PRD> ==>fix

<ret:BRN>701</ret:BRN> ==> retrieve from CustInfoService using AccountToDebit

<ret:TXNBRN>701</ret:TXNBRN> ==> make a call to CustInfoService using Account number

<ret:TXNACC>1055321777616</ret:TXNACC> ==> fetch from "AccountToDebit" in json request

<ret:TXNCCY>GHS</ret:TXNCCY> ==>fix

<ret:TXNAMT>1.00</ret:TXNAMT> ==> fetch from "Amount" in json request

<ret:NARRATIVE>GIFMIS Debit TransferP04001?</ret:NARRATIVE> ==> fetch from "TransactionDetails" in json request

<ret:SCODE>300323</ret:SCODE> ==> fix

<ret:Cheque-Details>

<ret:SRNO>240101</ret:SRNO> ==> fix

<ret:PAYBRN>701</ret:PAYBRN> ==> retrieved from CustInfoService using AccountToDebit

<ret:CHQNO>300311</ret:CHQNO> ==> fetch from "DestBankCode" in json request

<ret:EXPDT>9999-12-31</ret:EXPDT> ===> fix

<ret:BENFNAME>GIP TRANSFER</ret:BENFNAME> ==> fetch from "TransactionDetails" in json request

<ret:BENFADDR1>1030835430712</ret:BENFADDR1> ==> fetch from "AccountToCredit" in json request

</ret:Cheque-Details>

</ret:Transaction-Details>

</ret:FCUBS\_BODY>

</ret:CREATETRANSACTION\_FSFS\_REQ>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Body>

<CREATETRANSACTION\_FSFS\_RES xmlns="http://fcubs.ofss.com/service/FCUBSRTService">

<FCUBS\_HEADER>

<SOURCE>FLEXCUBE</SOURCE>

<UBSCOMP>FCUBS</UBSCOMP>

<MSGID>9241880685291459</MSGID>

<CORRELID>00430011003</CORRELID>

<USERID>PAPSSUSER</USERID>

<BRANCH>701</BRANCH>

<MODULEID>RT</MODULEID>

<SERVICE>FCUBSRTService</SERVICE>

<OPERATION>CreateTransaction</OPERATION>

<SOURCE\_USERID>PAPSSUSER</SOURCE\_USERID>

<DESTINATION>GIP</DESTINATION>

<MULTITRIPID>6241880887225719</MULTITRIPID>

<FUNCTIONID>DEGRTTLR</FUNCTIONID>

<ACTION>NEW</ACTION>

<MSGSTAT>SUCCESS</MSGSTAT>

</FCUBS\_HEADER>

<FCUBS\_BODY>

<Transaction-Details>

<XREF>101900066199</XREF>

<FCCREF>701PA02212280053</FCCREF>

<PRD>PA02</PRD>

<BRN>701</BRN>

<TXNBRN>701</TXNBRN>

<TXNACC>1055321777616</TXNACC>

<TXNCCY>GHS</TXNCCY>

<TXNAMT>1</TXNAMT>

<TXNTRN>PAP</TXNTRN>

<OFFSETBRN>000</OFFSETBRN>

<OFFSETACC>100131095</OFFSETACC>

<OFFSETCCY>GHS</OFFSETCCY>

<OFFSETAMT>1</OFFSETAMT>

<OFFSETTRN>PAP</OFFSETTRN>

<XRATE>1</XRATE>

<LCYAMT>1</LCYAMT>

<TXNDATE>2021-08-16</TXNDATE>

<VALDATE>2021-08-16</VALDATE>

<RELCUST>3217776</RELCUST>

<TCYCHGAMT>0</TCYCHGAMT>

<TCYTOTCHGAMT>0</TCYTOTCHGAMT>

<NARRATIVE>GIFMIS Debit TransferP04001?</NARRATIVE>

<MAKERID>PAPSSUSER</MAKERID>

<MAKERSTAMP>2021-08-16 12:55:53 PM</MAKERSTAMP>

<CHECKERID>PAPSSUSER</CHECKERID>

<CHECKERSTAMP>2021-08-16 12:55:53 PM</CHECKERSTAMP>

<RECSTAT>O</RECSTAT>

<AUTHSTAT>A</AUTHSTAT>

<LCYCHGAMT>0</LCYCHGAMT>

<DRACC>OFS</DRACC>

<TXNDRCR>C</TXNDRCR>

<BOOKDATE>2021-08-16</BOOKDATE>

<FT>N</FT>

<LCYTOTCHGAMT>0</LCYTOTCHGAMT>

<DENMCCY1>GHS</DENMCCY1>

<DENMAMT1>0</DENMAMT1>

<DENMAMT2>0</DENMAMT2>

<ACCTITLE1>WORD PROPERTIES COMPANY LIMITED-AGENT TRANSACTION ACCOUNT</ACCTITLE1>

<ACCTITLE2>PAPSS INBOUND</ACCTITLE2>

<ACTAMT>1</ACTAMT>

<Charge-Details>

<CHGCOMP>TXN\_FEE</CHGCOMP>

<CHGAMT>0</CHGAMT>

<CHGCCY>GHS</CHGCCY>

<MISHEAD>INCOME</MISHEAD>

<TXNCODE>PAP</TXNCODE>

<NETTINGIND>N</NETTINGIND>

<CHGGL>310006168</CHGGL>

<TYPE>F</TYPE>

<LCYCHG>0</LCYCHG>

<XRATE>1</XRATE>

</Charge-Details>

<Cheque-Details>

<SRNO>1</SRNO>

<CHQNO>300311</CHQNO>

<BENFNAME>GIP TRANSFER</BENFNAME>

<BENFADDR1>1030835430712</BENFADDR1>

</Cheque-Details>

</Transaction-Details>

<FCUBS\_WARNING\_RESP>

<WARNING>

<WCODE>GW-SAV-01</WCODE>

<WDESC>Transaction Saved Successfully</WDESC>

</WARNING>

</FCUBS\_WARNING\_RESP>

</FCUBS\_BODY>

</CREATETRANSACTION\_FSFS\_RES>

</S:Body>

</S:Envelope>

*\*\*\* If RT service return no response, call QueryTransaction method to check the true transaction status as below:*

CBS URL: <https://10.180.142.218:7081/FCUBSRTServiceV1.5>

Operation: QueryTransaction

*----------------------------Request----------------------------------------*

*<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:fcub="http://fcubs.ofss.com/service/FCUBSRTService">*

*<soapenv:Header>*

*<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" soapenv:mustUnderstand="1">*

*<wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="UsernameToken-1">*

*<wsse:Username>esbtestuser</wsse:Username>*

*<wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">fidelity,123</wsse:Password>*

*<wsse:Nonce EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">PCamrOP55EQorWOZ8zOBHA==</wsse:Nonce>*

*<wsu:Created>2015-10-28T02:45:11.593Z</wsu:Created>*

*</wsse:UsernameToken>*

*</wsse:Security>*

*</soapenv:Header>*

*<soapenv:Body>*

*<fcub:QUERYTRANSACTION\_IOFS\_REQ>*

*<fcub:FCUBS\_HEADER>*

*<fcub:SOURCE>YELLOSAVE</fcub:SOURCE>*

*<fcub:UBSCOMP>FCUBS</fcub:UBSCOMP>*

*<fcub:MSGID>YW3152455211</fcub:MSGID>*

*<fcub:CORRELID>YW3152455211</fcub:CORRELID>*

*<fcub:USERID>YSAVEUSER</fcub:USERID>*

*<fcub:BRANCH>083</fcub:BRANCH>*

*<fcub:MODULEID>RT</fcub:MODULEID>*

*<fcub:SERVICE>FCUBSRTService</fcub:SERVICE>*

*<fcub:OPERATION>QueryTransaction</fcub:OPERATION>*

*<fcub:SOURCE\_OPERATION>QueryTransaction</fcub:SOURCE\_OPERATION>*

*<fcub:SOURCE\_USERID>YSAVEUSER</fcub:SOURCE\_USERID>*

*<!--Optional:-->*

*<fcub:ADDL>*

*<!--Zero or more repetitions:-->*

*<fcub:PARAM>*

*<fcub:NAME>SERVERSTAT</fcub:NAME>*

*<fcub:VALUE>HOST</fcub:VALUE>*

*</fcub:PARAM>*

*</fcub:ADDL>*

*</fcub:FCUBS\_HEADER>*

*<fcub:FCUBS\_BODY>*

*<fcub:Transaction-Details-IO>*

*<fcub:XREF>YD34761672</fcub:XREF> ==> XREF value used for Create transaction*

*</fcub:Transaction-Details-IO>*

*</fcub:FCUBS\_BODY>*

*</fcub:QUERYTRANSACTION\_IOFS\_REQ>*

*</soapenv:Body>*

*</soapenv:Envelope>*

*and if in response ECODE =* ***WB-TXN-001***

*Thus: <ECODE>****WB-TXN-001****</ECODE> then response to Channel,*

{

“transactionStatus”: “Transaction has timeout, please try again”

}

*Else if in response MSGSTAT = SUCCESS proceed to point 6 -* Mapping of core banking response to GhIPSS request

*\*\*\* If RT service is successful (MSGSTAT = SUCCESS), proceed to point 6 "core banking response to GhIPSS request"*

=========== 6. Mapping of core banking response to GhIPSS request ==========

Response msg 🡺Mapping Core banking SOAP to GhIPSS SOAP Request

|  |  |  |
| --- | --- | --- |
| Core Banking SOAP | GhIPSS SOAP | values |
| TXNAMT | Amount | Format per GhIPSS spec  (12 digits, 000000001000=> ghs 10; 000000050000=>ghs500; 000000000100=> ghs 1) |
| No mapping | datetime | current time (YYMMDDHHMMSS) |
| No mapping | TrackingNum | Auto-generated (6 digits) |
| No mapping | FunctionCode | Default (240) |
| No mapping | OrigineBank | Default (300323) |
| CHQNO | DestBank |  |
| XREF | SessionID |  |
| No mapping | ChannelCode | Default (100) |
| TXNACC | AccountToDebit |  |
| ACCTITLE1 | NameToDebit |  |
| BENFADDR1 | AccountToCredit |  |
| BENFNAME | NameToCredit |  |
| NARRATIVE | Narration |  |
| No mapping | Login | Default (300323) |
| No mapping | Password | As shared to use |

=========== 7. IIB request to GhIPSS ==========

soapenv:Body>

<com:GIPTransactionOp>

<ReqGIPTransaction>

<Amount>1</Amount> ==> 12 digits, 000000001000=> ghs 10; 000000050000=>ghs500; 000000000100=> ghs 1 )

<datetime>230724180312</datetime> ==> current time (YYMMDDHHMMSS)

<TrackingNum>127288</TrackingNum>

<FunctionCode>240</FunctionCode>

<OrigineBank>300323</OrigineBank>

<DestBank>300323</DestBank>

<SessionID>101900066199</SessionID>

<ChannelCode>100</ChannelCode>

<AccountToDebit>1090033199418</AccountToDebit>

<NameToDebit>AUGSUTINE JESSON</NameToDebit> ==> (trim to 50 characters, replace special characters with white space)

<AccountToCredit>1030835430712</AccountToCredit>

<NameToCredit>JOHN MENSAH</NameToCredit>

<Narration>TEST</Narration>

<Login>300323</Login>

<Password>3104b15a6e5e57ee5663f64e6a1fb0d940d7824c875e42d35f9a30a72</Password>

</ReqGIPTransaction>

</com:GIPTransactionOp>

</soapenv:Body>

=========== 8. GhIPSS response to IIB ==========

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns2:GIPTransactionOpResponse xmlns:ns2="com.ghipss.gip">

<GIPTransactionResponse>

<Amount>000000000100</Amount>

<datetime>2407061803</datetime>

<TrackingNum>127288</TrackingNum>

<FunctionCode>240</FunctionCode>

<OrigineBank>300323</OrigineBank>

<DestBank>300311</DestBank>

<SessionID>101900066199</SessionID>

<NameToDebit>WORD PROPERTIES COMPANY LIMITED</NameToDebit>

<AccountToDebit>1055321777616</AccountToDebit>

<NameToCredit>JOHN MENSAH</NameToCredit>

<AccountToCredit>1030835430712</AccountToCredit>

<Narration>TEST TRANSACTION</Narration>

<OneTimePassword xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>

<Login xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>

<Password xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>

<ActCode>000</ActCode>

<AprvCode>0000</AprvCode>

</GIPTransactionResponse>

</ns2:GIPTransactionOpResponse>

</soap:Body>

</soap:Envelope>

</soapenv:Envelope>

*\*\*\* If in GhIPSS response, ActCode tag value = 000*

*Thus: <ActCode>000</ActCode>*

*then proceed to step # 9 then #10 as below*

=========== #9. Mapping GhIPSS response to Channel ==========

Request msg 🡺Mapping GhIPSS response to Channel REST

|  |  |
| --- | --- |
| GhIPSS - SOAP body | Channel - JSON body |
| SessionID | channelRef |
| Amount | amount |
| NameToCredit | nameToCredit |
| AccountToCredit | accountToCredit |
| Narration | transactionDetails |
| ActCode | statusCode |
| aprvCode | aprvCode |
|  |  |

=========== #10 IIB response to Channel ==========

{

    "channelRef": "101900166252",

"amount": "50",

    " accountToCredit": "1020820171412",

    " nameToCredit": "OLAM PURCHASE ACCOUNT",

"transactionDetail": "GIP PAYMENT FOR GOODS"

"statusCode": "000"

“aprvCode”: “000000”

}

*\*\*\* If in GhIPSS response, ActCode tag value in (100, 381)*

*Thus: <ActCode>100</ActCode>; <ActCode>381</ActCode>*

*call Reverse RT SERVICE as below:*

======== #11 fetch FCCREF value from successful CreateTransaction call=======

Thus: <FCCREF>701PA02212280053</FCCREF>

=============== #12 call ReverseTransaction call ===============

Request msg 🡺Mapping to call RT Service for reversal

|  |  |
| --- | --- |
| SOAP body | Remark |
| SOURCE | Default value: GIP |
| UBSCOMP | Dafault value: FCUBS |
| MSGID | Unique auto-generated values (max. of 18 characters) |
| CORRELID | Unique auto-generated values (max. of 18 characters) |
| USERID | Value in USERID tag |
| BRANCH | First 3 digits value in ‘FCCREF’ tag |
| MODULEID | Default value: RT |
| SERVICE | Default value: FCUBSRTService |
| OPERATION | Default value: CreateTransaction |
| SOURCE\_OPERATION | Default value: CreateTransaction |
| SOURCE\_USERID | Value in USERID tag |
| NAME | Default value: SERVERSTAT |
| VALUE | Default value: HOST |
| FCCREF | Value in FCCREF tag |
|  |  |

----------------------- Request from IIB to CBA – ***Reversal call***----------------------

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:fcub="http://fcubs.ofss.com/service/FCUBSRTService">

<soapenv:Header>

<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" soapenv:mustUnderstand="1">

<wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="UsernameToken-1">

<wsse:Username>esbtestuser</wsse:Username>

<wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">fidelity,123</wsse:Password>

<wsse:Nonce EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">PCamrOP55EQorWOZ8zOBHA==</wsse:Nonce>

<wsu:Created>2015-10-28T02:45:11.593Z</wsu:Created>

</wsse:UsernameToken>

</wsse:Security>

</soapenv:Header>

<soapenv:Body>

<fcub:REVERSETRANSACTION\_IOPK\_REQ>

<fcub:FCUBS\_HEADER>

<fcub:SOURCE>GIP</fcub:SOURCE>

<fcub:UBSCOMP>FCUBS</fcub:UBSCOMP>

<fcub:MSGID>99128250730178</fcub:MSGID>

<fcub:CORRELID>99128207360178</fcub:CORRELID>

<fcub:USERID>GIPUSER</fcub:USERID>

<fcub:BRANCH>701</fcub:BRANCH>

<fcub:MODULEID>RT</fcub:MODULEID>

<fcub:SERVICE>FCUBSRTService</fcub:SERVICE>

<fcub:OPERATION>ReverseTransaction</fcub:OPERATION>

<fcub:SOURCE\_OPERATION>ReverseTransaction</fcub:SOURCE\_OPERATION>

<fcub:SOURCE\_USERID>GIPUSER</fcub:SOURCE\_USERID>

<fcub:ADDL>

<!--Zero or more repetitions:-->

<fcub:PARAM>

<fcub:NAME>SERVERSTAT</fcub:NAME>

<fcub:VALUE>HOST</fcub:VALUE>

</fcub:PARAM>

</fcub:ADDL>

</fcub:FCUBS\_HEADER>

<fcub:FCUBS\_BODY>

<fcub:Transaction-Details-IO>

<fcub:FCCREF>701PA02212280053</fcub:FCCREF>

</fcub:Transaction-Details-IO>

</fcub:FCUBS\_BODY>

</fcub:REVERSETRANSACTION\_IOPK\_REQ>

</soapenv:Body>

</soapenv:Envelope>

-------------- Response from CBA to IIB – ***Reversal call***----------------------

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Body>

<REVERSETRANSACTION\_IOPK\_RES xmlns="http://fcubs.ofss.com/service/FCUBSRTService">

<FCUBS\_HEADER>

<SOURCE>FLEXCUBE</SOURCE>

<UBSCOMP>FCUBS</UBSCOMP>

<MSGID>9243590686483705</MSGID>

<CORRELID>99128207360178</CORRELID>

<USERID>GIPUSER</USERID>

<BRANCH>701</BRANCH>

<MODULEID>RT</MODULEID>

<SERVICE>FCUBSRTService</SERVICE>

<OPERATION>ReverseTransaction</OPERATION>

<SOURCE\_USERID>GIPUSER</SOURCE\_USERID>

<DESTINATION>GIP</DESTINATION>

<MULTITRIPID>6243590888420376</MULTITRIPID>

<FUNCTIONID>DEGRTTLR</FUNCTIONID>

<ACTION>REVERSE</ACTION>

<MSGSTAT>SUCCESS</MSGSTAT>

</FCUBS\_HEADER>

<FCUBS\_BODY>

<FCUBS\_WARNING\_RESP>

<WARNING>

<WCODE>GW-REV-01</WCODE>

<WDESC>Transaction Reversed Succesfully</WDESC>

</WARNING>

</FCUBS\_WARNING\_RESP>

</FCUBS\_BODY>

</REVERSETRANSACTION\_IOPK\_RES>

</S:Body>

</S:Envelope>

*and if in response MSGSTAT tag = SUCCESS then response to Channel,*

{

“transactionStatus”: ““Transaction has failed, please try again”

}

*Else if in response MSGSTAT = FAILURE then response to Channel,*

{

“transactionStatus”: “Contact the customer care team”

}

*\*\*\* If in GhIPSS response, ActCode value in (909, 912, NULL, 990, 911, 999, timeouts) then proceed to below:*

==================== Invoke Transaction Status Query call ===========

Request msg 🡺Mapping GhIPSS response to Transaction Status Query(TSQ) Request

|  |  |  |
| --- | --- | --- |
| GhIPSS Response | TSQ Request | values |
| Amount | Amount | as return in GhIPSS response |
| datetime | datetime | as return in GhIPSS response |
| TrackingNum | TrackingNum | as return in GhIPSS response |
| FunctionCode | FunctionCode | Default (111) |
| OrigineBank | OrigineBank | as return in GhIPSS response |
| DestBank | DestBank | as return in GhIPSS response |
| SessionID | SessionID | as return in GhIPSS response |
| ChannelCode | ChannelCode | as return in GhIPSS response |
| AccountToDebit | AccountToDebit | as return in GhIPSS response |
| NameToDebit | NameToDebit | as return in GhIPSS response |
| AccountToCredit | AccountToCredit | as return in GhIPSS response |
| NameToCredit | NameToCredit | as return in GhIPSS response |
| Narration | Narration | as return in GhIPSS response |
| Login | Login | Default (300323) |
| Password | Password | As shared to use |

**Request:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:com="com.ghipss.gip">

<soapenv:Header>

<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" soapenv:mustUnderstand="1">

<wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="UsernameToken-1">

<wsse:Username>esbtestuser</wsse:Username>

<wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">fidelity,123</wsse:Password>

<wsse:Nonce EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">PCamrOP55EQorWOZ8zOBHA==</wsse:Nonce>

<wsu:Created>2020-10-28T02:45:11.593Z</wsu:Created>

</wsse:UsernameToken>

</wsse:Security>

</soapenv:Header>

<soapenv:Body>

<com:GIPTransactionOp>

<ReqGIPTransaction>

<Amount>000000000200</Amount>

<datetime>220719172647</datetime>

<TrackingNum>395725</TrackingNum>

<FunctionCode>111</FunctionCode>

<OrigineBank>300595</OrigineBank>

<DestBank>300315</DestBank>

<SessionID>071999695281</SessionID>

<ChannelCode>100</ChannelCode>

<NameToDebit>Hayford Owusu Brobbey</NameToDebit>

<AccountToDebit>0246089019</AccountToDebit>

<NameToCredit>ENOCH DANSO CLINTON</NameToCredit>

<AccountToCredit>0246089019000</AccountToCredit>

<Narration>test payment</Narration><Login>300323</Login>

<Password>c941a4ec7bd7937a58c1e6aae3dd50160dcc14d6f10c754041d7b74f9a84d2</Password>

</ReqGIPTransaction>

</com:GIPTransactionOp>

</soapenv:Body>

</soapenv:Envelope>

------------------Response-------------------------------

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Body>

<ns2:GIPTransactionOpResponse xmlns:ns2="com.ghipss.gip">

<GIPTransactionResponse>

<Amount>000000000200</Amount>

<datetime>220719172647</datetime>

<TrackingNum>395725</TrackingNum>

<FunctionCode>111</FunctionCode>

<OrigineBank>300595</OrigineBank>

<DestBank>300315</DestBank>

<SessionID>071999695281</SessionID>

<ChannelCode>100</ChannelCode>

<NameToDebit>Hayford Owusu Brobbey</NameToDebit>

<AccountToDebit>0246089019</AccountToDebit>

<NameToCredit>ENOCH DANSO CLINTON</NameToCredit>

<AccountToCredit>0246089019000</AccountToCredit>

<Narration>test payment</Narration>

<OneTimePassword xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>

<ActCode>000</ActCode>

<AprvCode>000000</AprvCode>

<StatusQuery>000</StatusQuery>

</GIPTransactionResponse>

</ns2:GIPTransactionOpResponse>

</S:Body>

</S:Envelope>

*and if in response ActCode tag = 000 and StatusQuery tag = 000 then response to Channel,*

{

    "channelRef": "101900166252",

"amount": "50",

    " accountToCredit": "1020820171412",

    " nameToCredit": "OLAM PURCHASE ACCOUNT",

"transactionDetail": "GIP PAYMENT FOR GOODS"

"statusCode": "000"

“aprvCode”: “0000000

}

*Else if in response ActCode tag not 000 or StatusQuery tag not 000 then response to Channel,*

{

“transactionStatus”: “Contact the customer care team”

}