Carnelian Web Service Documentation

(Foreign Office Transaction Create Request

Foreign Office Transaction Detail inquiry)

Version 2.5

REVISION HISTORY

Version	Amendment	Ref in Doc	Updated By	Date
1.0	All	All	SXI	April 19, 2012
1.2	Added input/output parameters	All	SXI	May 2, 2012
1.3	Added : New fields (last name, first name, middle name) for both remitter and beneficiary Define date format (mm/dd/yyyy) Transaction detail inquiry - Added new input parameter field, remittance date to and date from - Added mandatory handling, date and application number, either should be available Transaction create request - Added Validation against application parameter for multiple transactions - Assigned single cover number for multiple transactions in single request - Description for beneficiary customer type - Bank code, bank name and branch code – added mandatory handling	All	SXI	May 3, 2012
1.4	Revised mandatory validation from the input/output parameter Added new fields in the input/output parameters		SXI	May 17, 2012
1.5	Transaction create Request WS a. Add remitter continent and remitter office phone as input parameter b. Add process flow diagram c. Add sample xml request response Transaction inquiry request WS a. Add input parameters: cover number, application number and transaction feedback status b. Add sample xml request response		SXI	June 6, 2012
1.6	Transaction create Request WS – revised the display of settlement amount field Added encryption algorithm Conclusion – added security features and logs…		SXI	June 15, 2012
1.7	Add items requested from security Remove mandatory for institution code and subscriber reference number		SXI	June 21, 2012
1.8	Added new error messages.		Melanie Balao-as	April 28, 2013

Version	Amendment	Ref in Doc	Updated By	Date
1.9	1) Added VPN parameters and form 2) Added Carnelian Web Service Documentation for Clients		Melanie Balao-as	August 13, 2013
1.10	Update Transaction Create Request sample XML		Korina del Castillo	August 30, 2013
1.11	Changed WS URL Removed Remitter ID Number1 in Transaction Create (duplicate)		Melanie Balao-as	September 17, 2013
2.0	Transaction detail inquiry 1) Added new input parameter field, remittance date to and date from 2) Included FSD addendum version 1.2		SXI	September 30, 2013
	Transaction detail create & inquiry 1) Created a column to indicate if a field is mandatory or optional		Korina del Castillo	
2.1	Added Corresponding Error Code for the Web Service Error Message Table		Korina del Castillo	November 18, 2013
2.2	1) Modified character length of Beneficiary First Name to 100 characters. 2) Set Beneficiary/Remitter First Name and Middle Name to Mandatory.		Korina del Castillo	June 2, 2014
2.3	Changed CRS TEST URL to https://172.20.20.31:8181/carnelian-web-service-server/cxf/remittance		Melanie Balao-as	May 20, 2015
2.4	Changed CRS PROD URL to https://172.20.40.240:8181/carnelian-web-service-server/cxf/remittance		Melanie Balao-as	August 09, 2015
2.5	Changed CRS TEST URL to https://172.20.20.249:8181/carnelian-web-service-server/cxf/remittance		Melanie Balao-as	April 14, 2016

Acceptance

We hereby signify our satisfaction and acceptance of this Functional Specifications Document as a description of the functionality of the Web Service Module for Business Outlet Partners.

This Functional Specifications Document supersedes any previous document issued on the requirements for the project.

SIGNATORIES:

Prepared By:

China Bank Melanie N. Balao-as ________________

Reviewed By:

Tie-up Representative ________________

Tie-up Representative ________________

Tie-up Representative ________________

Introduction

This document aims to discuss how Carnelian Remittance System hosts its web services and its security mechanisms. This will describe the different security mechanisms that are used, and the different components that are needed to make it work.

Web Service Library – Apache CXF

The web service library of choice was Apache CXF which can be found on (http://cxf.apache.org). CXF is an open source web library that implements different specifications and standards. Here are some of the standards were implemented:

SOAP

WSI Basic Profile

WSDL

WS-Addressing

WS-Reliable Message

WS-Security

WS-Secure Conversation

WS-Trust

As with every web service, there is a client and a server. A WSDL file will also be provided for generating the proper client.

Web Service Security

Security is one very important aspect for web services. There exist several reasons why we will need security and here are some of the following:

- Communication between client and server should not be viewable by a third party as it travels across the wire.
- Server should be able to determine from whom the message was coming from and be able to verify that the sender was who he/she claims to be.
- Customer needs to able to ensure that the data being transmitted was not tampered with.

Security Framework – WSS4J

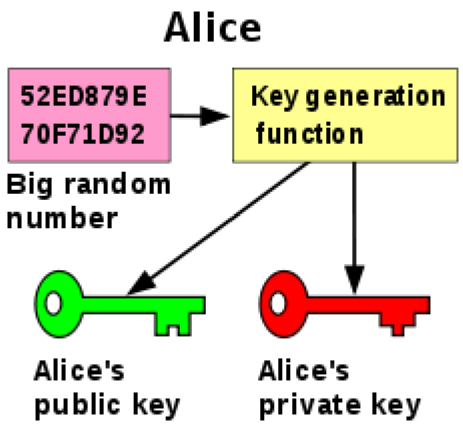
The security framework used in our case is WSS4J (Web Service Security For Java) which can be found on http://ws.apache.org/wss4j/. It is an implementation of the OASIS Web Services Security (WS-Security). It is a library that can be used to sign and verify SOAP messages. WS-Security is a standard that is used by many web service implementations.

The following standards are implemented.

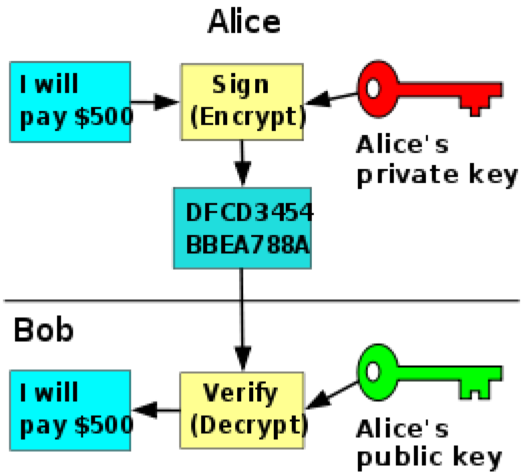
- Web Services Security : SOAP Message Security 1.1
- Username Token profile 1.1
- X.509 Certificate Token Profile 1.1

Security Mechanisms – Taken From (http://cxf.apache.org/docs/ws-security.html)

- ***Transport Layer Security*** – use of SSL Certificates to handle transport layer security.
- ***End to End Security*** – security is handled via public key encryption. With public key cryptography, a user has a pair of public and private keys. These are generated using a large prime number and a key function. The keys are related mathematically, but cannot be derived from one another. With these keys we can encrypt messages. For example, if Bob wants to send a message to Alice, he can encrypt a message using her public key. Alice can then decrypt this message using her private key. Only Alice can decrypt this message as she is the only one with the private key.



- ***Message Signing*** – Messages can also be signed. This allows you to ensure the authenticity of the message. If Alice wants to send a message to Bob, and Bob wants to be sure that it is from Alice, Alice can sign the message using her private key. Bob can then



verify that the message is from Alice by using her public key.

- ***Message Encryption***
- ***Passsword Authentication Token*** -WS-Security supports many ways of specifying tokens. One of these is the UsernameToken header. It is a standard way to communicate a username and password or password digest to another endpoint. Passwords can also expire for higher security.

Encryption Algorithm: SHA256 or higher

Salt size: 8 bytes

Iterations: at least 1000 times (as specified) in the Jasypt reference site

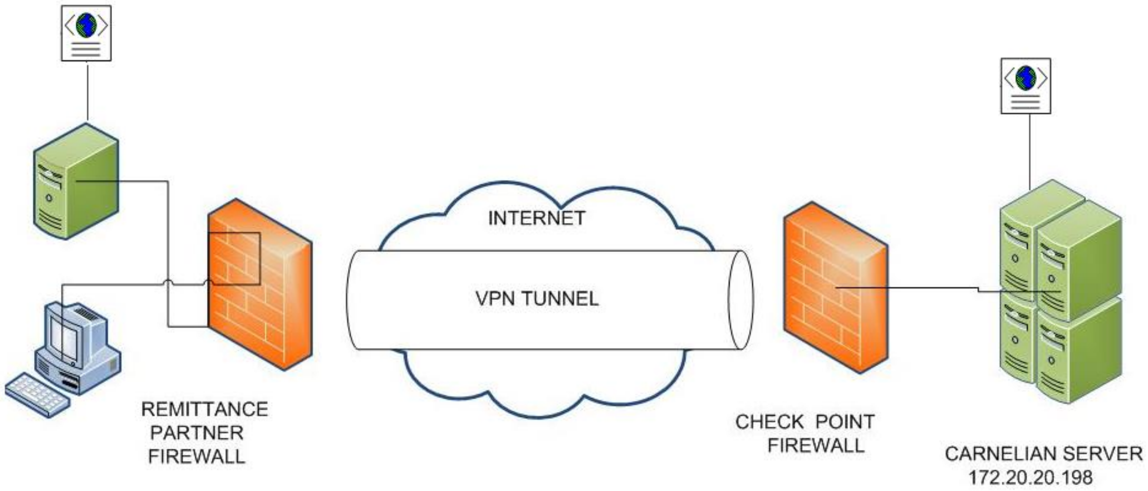
Conclusion

Carnelian web services adhere to the WS-Security standard which is a standard across web service implementations. This would ensure that conversation between the client and the server are secure and reliable.

Also, all existing security features and logs (i.e audit log report and server logs) in existing Carnelian Application will also be applied to Carnelian web services

Reference Site for Jasypt: http://jasypt.org/

SYSTEM ARCHITECTURE:



Kindly fill-up Remittance Partner part.

VPN FORM

CHINA BANK			***REMITTANCE PARTNER***	
VPN Information			**VPN Information**	
Supplier	**CheckPoint**		Supplier	
Type	**CheckPoint NGX**		Type	
Model	**R75.4**		Model	
OS	**Secure Platform**		OS	
Public IP Peer Address	**66.198.139.189**		Public IP Peer Address	

IKE Proposal Parameters			**IKE Proposal Parameters**	
Proposal Name	IKE-3DES-MD5		Proposal Name	
Authentication Mode	Pre-shared Key		Authentication Mode	
Preshared Key	Send via email		Preshared Key	
Authentication Algorithm	MD5		Authentication Algorithm	
Encryption Algorithm	3DES		Encryption Algorithm	
Diffie-Hellman Group	Group 2		Diffie-Hellman Group	
Lifetime Measurement	Time		Lifetime Measurement	
Data Lifetime	10000		Data Lifetime	
Time Lifetime	86400		Time Lifetime	

IPSEC Parameters			**IPSEC Parameters**	
Authentication Algorithm	MD5		Authentication Algorithm	
Encryption Algorithm	3DES		Encryption Algorithm	
Encapsulation Mode	Tunnel (LAN to LAN)		Encapsulation Mode	
Perfect Forward Secrecy	Disabled		Perfect Forward Secrecy	
Lifetime Measurement	Time		Lifetime Measurement	
Data Lifetime	10000		Data Lifetime	
Time Lifetime	28800		Time Lifetime	

CARNELIAN WEB SERVICE DOCUMENTATION – FOR CLIENTS

Introduction:

This document lists the things that will be needed to connect to the Carnelian Web Service.

Pre-requisites:

Things that are needed by a client connecting to Carnelian Web Services

- ***WSDL file*** – this will contain the meta information regarding the web service.
- ***Web Service Endpoint Address*** – represents that address of the web service to which the client will connect to.
  CRS Web Service Test URL is
  https://172.20.20.249:8181/carnelian-web-service-server/cxf/remittance
  CRS Web Service Prod URL is
  https://172.20.40.240:8181/carnelian-web-service-server/cxf/remittance
- ***Certificate*** for connecting to https – the certificate that needs to be used in order to connect to an https service.
- ***Provided Username/Password*** – the username/password combination for additional authentication to Carnelian Remittance Web Service. *To be provided once 3rd Party Agreement has been signed off.*
- ***Password Encryption Library*** –http://www.jasypt.org/encrypting-passwords.html. We are currently using the StrongPasswordEncyptor class.

Generate web service client from a WSDL file.

- A client can generate the needed web service from a WSDL file that will be provided.
- He/She can use Eclipse/Netbeans or any library for that matter to generate a web service.
- Since the connection is using an https connection, he or she will need to be able to import the certificate in order for his web service to use it.

Using a Certificate

- A certificate will be given to the client in order for him to connect to an https service. The certificate will be given. The certificate needs to be imported into the keystore or to be referenced.
- Certificate can be imported into a keystore (Java) and be reference by the application.
- The keystore needs to be referenced by the application server in order for the web service to use it properly.

Sample Command for importing a certificate into a keystore.

```
C:\>keytool -importcert -file <file_name> -keystore <keystore_name>
```

Transaction Create Request

This facility allow user to request for create remittance transaction in Carnelian (CBC). User can send one or multiple transactions in a single request.

Note:

For multiple transactions send in single request:

a. System validate, if number of transaction is greater than max number of allowed transactions defined in application parameter, system will reject the request.
b. System to assign cover number for the multiple/single transactions received.

Parameter Name	Data Type	Size	Parameter Description	
Input Parameters				
Header Section				
Username (Required)	String	10	Username that a client will use to authenticate himself inside carnelian.	MANDATORY
Password (Required)	String	---	An encrypted password that will be sent. Jasypt is used for this purpose.	MANDATORY
Detail Section				
Transaction Date	Date (MM/DD/YYYY)	10	Foreign Office date	MANDATORY
Remitter ID	String	32	User defined Remitter id	OPTIONAL
Remitter Last Name (Required)	String	100	Remitter Last Name	MANDATORY
Remitter First Name	String	100	Remitter First Name	MANDATORY
Remitter Middle Name	String	100	Remitter Middle Name	MANDATORY
Remitter Customer Type	String	1	Remitter Customer Type “I” – Individual “C” - Corporate	OPTIONAL
Remitter Address1	String	100	Remitter Address – No and Street	OPTIONAL
Remitter Address2	String	100	Remitter Address - town/city	OPTIONAL
Remitter Address3	String	100	Remitter Address - State	OPTIONAL
Remitter Address4	String	35	Remitter Address - Country	OPTIONAL
Remitter PO Box	String	20	Beneficiary PO Box	OPTIONAL

Parameter Name	Data Type	Size	Parameter Description	
Remitter Office Phone Number	String	100	Remitter Office Phone Number	OPTIONAL
Remitter Email Address	String	250	Remitter Email Address	OPTIONAL
Remitter ID Type1	String	10	Remitter ID Type1	OPTIONAL
Remitter ID Number1	String	150	Remitter ID Number1	OPTIONAL
Remitter ID Issue At 1	String	20	Remitter ID Issue At 1	OPTIONAL
Remitter ID Expiry Date1	Date (MM/DD/YYYY)	10	Remitter ID Expiry Date1	OPTIONAL
Remitter ID Type2	String	10	Remitter ID Type2	OPTIONAL
Remitter ID Number2	String	150	Remitter ID Number2	OPTIONAL
Remitter ID Issue At 2	String	20	Remitter ID Issue At 2	OPTIONAL
Remitter ID Expiry Date2	Date (MM/DD/YYYY)	10	Remitter ID Expiry Date2	OPTIONAL
Remitter Notification Type	Number	2	Valid Values: 1- Email, 2- Mobile, 3 – Fax	OPTIONAL
Remitter Zip Code	Number	10	Remitter Zip Code	OPTIONAL
Remitter Birthday	Date (MM/DD/YYYY)	10	Remitter Birthday	OPTIONAL
Remitter Nationality	String	30	Remitter Nationality	OPTIONAL
Remitter Gender	String	5	Remitter gender Values are : "M" – Male "F" - Female	OPTIONAL
Remitter Civil Status	String	5	Remitter civil status Values are: "S" – Single "M" – Married "W" – Widow/Widower "D" - Divorced	OPTIONAL
Remitter Mobile No (optional)	String	100	Remitter Mobile No	OPTIONAL
Remitter TIN	Numeric	20	Remitter TIN	OPTIONAL
Remitter Continent	String	32	Remitter continent	OPTIONAL
Remitter Office	String	100	Remitter Office Phone number	OPTIONAL

Parameter Name	Data Type	Size	Parameter Description	
Phone				
Beneficiary ID (optional)	String	100	User defined beneficiary id	OPTIONAL
Beneficiary Last Name (Required)	String	100	Beneficiary Last Name	MANDATORY
Beneficiary First Name	String	100	Beneficiary First Name	MANDATORY
Beneficiary Middle Name	String	100	Beneficiary Middle Name	MANDATORY
Beneficiary Customer Type	String	100	Beneficiary Customer Type "I" – Individual "C" – Corporate	OPTIONAL
Beneficiary Address1 (Required)	String	100	Beneficiary Address - Number and Street Name	MANDATORY for Door-to-Door Transactions
Beneficiary Address2	String	30	Beneficiary Address - town/city	OPTIONAL
Beneficiary Address3	String	40	Beneficiary Address – State/Province	OPTIONAL
Beneficiary Country	String	20	Beneficiary Address - Country	OPTIONAL
LandMark	String	40	LandMark	OPTIONAL
Beneficiary Profession	String	30	Beneficiary Profession	OPTIONAL
Beneficiary PO Box	String	10	Beneficiary PO Box	OPTIONAL
Beneficiary Office Phone Number	String	100	Beneficiary Office Phone Number	OPTIONAL
Beneficiary Email	String	250	Beneficiary Email	OPTIONAL
Beneficiary Notification Type	Number	2	valid values: 1-Email, 2-Mobile, 3-Fax	OPTIONAL
Beneficiary Zip Code	Number	10	Beneficiary Zip Code	OPTIONAL
Beneficiary Birthday	Date (MM/DD/YYYY)	10	Beneficiary Birthday	OPTIONAL
Beneficiary Nationality	String	30	Beneficiary Nationality	OPTIONAL
Beneficiary Gender	String	5	Beneficiary Gender "M" – Male "F" – Female	OPTIONAL
Beneficiary Civil Status	String	2	Beneficiary Civil Status	OPTIONAL

Parameter Name	Data Type	Size	Parameter Description	
Beneficiary Mobile No	String	40	Beneficiary Mobile No	OPTIONAL
Beneficiary TIN	Numeric	20	Beneficiary TIN	
Beneficiary Relation to Remitter	String	30	Beneficiary relation to the remitter	OPTIONAL
Alternate Recipient’s Relation to Beneficiary	String	50	Beneficiary relation to the specified alternate recipient	OPTIONAL
Application Number (Required)	String	40	Unique Application number of the foreign office	MANDATORY
Service Code	Numeric	10	Valid settlement mode: 01 – Credit to Account 02 – Credit to Other Bank 03 – CBC Branch Pickup 04 – Door-to-Door Cash 05 – Door-to-Door Product 06 – Account Opening 07 – Bills Payment 08 – Outlet Pickup 09 – Pickup Anywhere 10 – Open Card 11 – Cash Card	MANDATORY
Account Number	String	35	Account Number	MANDATORY for Credit to Account(CTA) and Credit to Other Bank(CTOB) transactions only
Bank Code	String	15	Bank Code Mandatory for Credit to Account and Credit to Other Banks Default to 'CBC' or 'CBS' if CTA For CTOB, system to validate against Other Bank Code table maintained in carnelian	MANDATORY for Credit to Account (CTA) and Credit to Other Bank (CTOB) transactions only
Branch Code	String	35	Branch Code Mandatory for Branch Pickup. System to validate against Branch Code table maintained in carnelian	MANDATORY for CBC Branch Pickup
Account Type	Numeric	20	Account type Values are: 1- CA, 2- SA, 3- TD, 4- OTRC, 5- Smart Money	MANDATORY for OTRC Account Opening transactions
Account Name	String	200	Account Name	OPTIONAL

Parameter Name	Data Type	Size	Parameter Description	
Branch Name	String	40	Branch Name	OPTIONAL
Funding Currency (Required)	Numeric	3	Funding Currency	MANDATORY
Funding Amount	Float		Amount of remittance in Funding Currency	MANDATORY if Settlement Amount field is blank
Counter Charge	Float		Service fee to be paid by the remitter to foreign office in payment currency	MANDATORY Default value is 0
Is Waived	Numeric	2	Service fee is waivable. Values are: 0-No 1-Yes	MANDATORY Default value is 0
Other Charges	Float		Other fees to be paid by remitter to foreign office in payment currency in payment currency	MANDATORY Default value is 0
Settlement Currency (Required)	String	3	Currency of the remittance to be received by the beneficiary	MANDATORY
Settlement Amount	Float		Allow tie-ups to input the Beneficiary Amount if Beneficiary Currency is given.	MANDATORY if Funding Amount field is blank.
Outlet Code	String	50	Outlet Code - mandatory for outlet pick up service code	MANDATORY for Outlet Pickup transactions
Bank Name	String	180	Bank name Mandatory for Credit to Other Banks and Credit to Account Default to "China Banking Corporation" or China Bank Savings Inc" for CTA For CTOB, system to validate against Other Bank Code table maintained in carnelian	MANDATORY for Credit to Account(CTA) and Credit to Other Bank(CTOB) transactions only
Gold Card Number	String	40	Mandatory for OTRC and Cash Card Opening	MANDATORY for OTRC Account Opening and Cash Card transactions
Institution/Biller Code	String	10	For Bills Payment service code (required)	MANDATORY for Bills Payment service code
Subscriber Reference Number	String	20	For Bills Payment service code (required)	MANDATORY for Bills Payment

Parameter Name	Data Type	Size	Parameter Description	
				service code
Subscriber Name	String	20	For Bills Payment service code (required)	MANDATORY for Bills Payment service code
Buying Rate	Float		Funding Currency to Settlement Currency	MANDATORY Default value is 1.00
Bills Payment Field 4	String	20	Bills Payment Field 4	OPTIONAL
Bills Payment Field 5	String	20	Bills Payment Field 5	OPTIONAL
Outlet Branch Code	String	10	Outlet Branch Code	OPTIONAL
Alternate Beneficiary Name	String	50	Alternate Beneficiary Name	OPTIONAL
Alternate Beneficiary relationship to Beneficiary	String	30	Alternate Beneficiary relationship to Beneficiary	OPTIONAL
Message to Beneficiary	String	255	Message to Beneficiary	OPTIONAL
Receiver’s Correspondent Bank	String	50	for SWIFT transactions	MANDATORY for SWIFT transactions
Sender’s Correspondent Bank	String	50	for SWIFT transactions	MANDATORY for SWIFT transactions
Sending Bank	String	50	for SWIFT transactions	MANDATORY for SWIFT transactions
Receiving Bank	String	50	for SWIFT transactions	MANDATORY for SWIFT transactions
Mode of Charge	String	3	for SWIFT transactions Valid Values: ‘SHA’-both front-end charge and back-end charge, ‘OUR’-frontend charge, ‘BEN’-backend charge	MANDATORY for SWIFT transactions
Purpose Code	String	20	for SWIFT transactions	MANDATORY for SWIFT transactions
Individual Code	String	20	for SWIFT transactions	MANDATORY for SWIFT transactions
Output Parameters				

Parameter Name	Data Type	Size	Parameter Description
Header Section			
Is Error	Integer		0 – no error occurred. 1 – Error occurred.
Error Message	String		An error message in case an error has occurred. Refer to annex B for the complete list of error codes
No Txn Recvd	Numeric		Number of Transactions Received
Tot Amt Recvd	Numeric		Total Amounts Received
Cover number	String		System generated cover number

Sample Request:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://remittance.webservice.carnelian.ccti.com/">
<soapenv:Header/>
<soapenv:Body>
<com:createTransaction soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<arg0>
<username xsi:type="xsd:string"><value></username>
<password xsi:type="xsd:string"><value></password>
</arg0>
**********record 1**************
<arg1>
<dtRemittance xsi:type="xsd:string"><value></dtRemittance>
<remitterId xsi:type="xsd:string"><value></remitterId>
<remitterLastName xsi:type="xsd:string"><value></remitterLastName>
<remitterFirstName xsi:type="xsd:string"><value></remitterFirstName>
<remitterMiddleName xsi:type="xsd:string"><value></remitterMiddleName>
<remitterCustType xsi:type="xsd:string"><value></remitterCustType>
<remitterAddr1 xsi:type="xsd:string"><value></remitterAddr1>
<remitterAddr2 xsi:type="xsd:string"><value></remitterAddr2>
<remitterAddr3 xsi:type="xsd:string"><value></remitterAddr3>
<remitterCountry xsi:type="xsd:string"><value></remitterCountry>
<remitterPoBox xsi:type="xsd:string"><value></remitterPoBox>
<remitterEmailAddr xsi:type="xsd:string"><value></remitterEmailAddr>
<remitterIdType1 xsi:type="xsd:string"><value></remitterIdType1>
<remitterIdNumber1 xsi:type="xsd:string"><value></remitterIdNumber1>
<remitterIdIssueAt1 xsi:type="xsd:string"><value></remitterIdIssueAt1>
<remitterExpiryDate1 xsi:type="xsd:string"><value></remitterExpiryDate1>
<remitterIdType2 xsi:type="xsd:string"><value></remitterIdType2>
<remitterIdNumber2 xsi:type="xsd:string"><value></remitterIdNumber2>
<remitterIdIssueAt2 xsi:type="xsd:string"><value></remitterIdIssueAt2>
<remitterExpiryDate2 xsi:type="xsd:string"><value></remitterExpiryDate2>
<remitterNotificationType><value></remitterNotificationType>
<remitterZipCode><value></remitterZipCode>
<remitterBdate xsi:type="xsd:string"><value></remitterBdate>
<remitterNationality xsi:type="xsd:string"><value></remitterNationality>
<remitterGender xsi:type="xsd:string"><value></remitterGender>
<remitterCivilStatus xsi:type="xsd:string"><value></remitterCivilStatus>
<remitterMobileNumber xsi:type="xsd:string"><value></remitterMobileNumber>
<remitterTin><value></remitterTin>
<remitterContinent xsi:type="xsd:string"><value></remitterContinent>
<remitterOfficePhoneNumber xsi:type="xsd:string"><value></remitterOfficePhoneNumber>
<beneId xsi:type="xsd:string"><value></beneId>
<beneLastName xsi:type="xsd:string"><value></beneLastName>
<benefirstName xsi:type="xsd:string"><value></benefirstName>
<beneMiddleName xsi:type="xsd:string"><value></beneMiddleName>
<beneCustType xsi:type="xsd:string"><value></beneCustType>
<beneAddr1 xsi:type="xsd:string"><value></beneAddr1>
<beneAddr2 xsi:type="xsd:string"><value></beneAddr2>
<beneAddr3 xsi:type="xsd:string"><value></beneAddr3>
<beneCountry xsi:type="xsd:string"><value></beneCountry>
<beneLandmark xsi:type="xsd:string"><value></beneLandmark>
<beneProfession xsi:type="xsd:string"><value></beneProfession>
<benePoBox xsi:type="xsd:string"><value></benePoBox>
<beneOfficePhoneNumber xsi:type="xsd:string"><value></beneOfficePhoneNumber>
<beneEmail xsi:type="xsd:string"><value></beneEmail>
<beneNotificationType xsi:type="xsd:string"><value></beneNotificationType>
<beneZipCode xsi:type="xsd:string"><value></beneZipCode>
<beneBdate xsi:type="xsd:string"><value></beneBdate>
<beneNationality xsi:type="xsd:string"><value></beneNationality>
```

```
<beneGender xsi:type="xsd:string"><value></beneGender>
<beneCivilStatus xsi:type="xsd:string"><value></beneCivilStatus>
<beneMobileNo xsi:type="xsd:string"><value></beneMobileNo>
<beneTin xsi:type="xsd:string"><value></beneTin>
<beneRelationToRemitter xsi:type="xsd:string"><value></beneRelationToRemitter>
<altRecRelToBene xsi:type="xsd:string"><value></altRecRelToBene>
<applicationNumber xsi:type="xsd:string"><value></applicationNumber>
<serviceCode><value></serviceCode>
<accountNumber xsi:type="xsd:string"><value></accountNumber>
<bankCode xsi:type="xsd:string"><value></bankCode>
<branchCode xsi:type="xsd:string"><value></branchCode>
<accountType><value></accountType>
<accountName xsi:type="xsd:string"><value></accountName>
<branchName xsi:type="xsd:string"><value></branchName>
<xmitCcyCd><value></xmitCcyCd>
<xmitAmount><value></xmitAmount>
<counterCharge><value></counterCharge>
<isWaived><value></isWaived>
<otherCharges><value></otherCharges>
<beneCcyCd xsi:type="xsd:string"><value></beneCcyCd>
<beneAmtConv><value></beneAmtConv>
<outletCd xsi:type="xsd:string"><value></outletCd>
<bankName xsi:type="xsd:string"><value></bankName>
<cardNo xsi:type="xsd:string"><value></cardNo>
<paymentField1 xsi:type="xsd:string"><value></paymentField1>
<paymentField2 xsi:type="xsd:string"><value></paymentField2>
<paymentField3 xsi:type="xsd:string"><value></paymentField3>
<rtBuyingApplied><value></rtBuyingApplied>
<paymentField4 xsi:type="xsd:string"><value></paymentField4>
<paymentField5 xsi:type="xsd:string"><value></paymentField5>
<outletBranchCd xsi:type="xsd:string"><value></outletBranchCd>
<altBeneName xsi:type="xsd:string"><value></altBeneName>
<altBeneRelToBene xsi:type="xsd:string"><value></altBeneRelToBene>
<messageToBene xsi:type="xsd:string"><value></messageToBene>
<receiverCorrBank xsi:type="xsd:string"><value></receiverCorrBank>
<senderCorrBank xsi:type="xsd:string"><value></senderCorrBank>
<sendingBank xsi:type="xsd:string"><value></sendingBank>
<receivingBank xsi:type="xsd:string"><value></receivingBank>
<modeOfCharge xsi:type="xsd:string"><value></modeOfCharge>
<purposeCode xsi:type="xsd:string"><value></purposeCode>
<individualCode xsi:type="xsd:string"><value></individualCode>
</arg1>
```

**********record 2**************

```
<arg1>
<dtRemittance xsi:type="xsd:string"><value></dtRemittance>
<remitterId xsi:type="xsd:string"><value></remitterId>
<remitterLastName xsi:type="xsd:string"><value></remitterLastName>
<remitterFirstName xsi:type="xsd:string"><value></remitterFirstName>
<remitterMiddleName xsi:type="xsd:string"><value></remitterMiddleName>
<remitterCustType xsi:type="xsd:string"><value></remitterCustType>
<remitterAddr1 xsi:type="xsd:string"><value></remitterAddr1>
<remitterAddr2 xsi:type="xsd:string"><value></remitterAddr2>
<remitterAddr3 xsi:type="xsd:string"><value></remitterAddr3>
<remitterCountry xsi:type="xsd:string"><value></remitterCountry>
<remitterPoBox xsi:type="xsd:string"><value></remitterPoBox>
<remitterEmailAddr xsi:type="xsd:string"><value></remitterEmailAddr>
<remitterIdType1 xsi:type="xsd:string"><value></remitterIdType1>
<remitterIdNumber1 xsi:type="xsd:string"><value></remitterIdNumber1>
<remitterIdIssueAt1 xsi:type="xsd:string"><value></remitterIdIssueAt1>
<remitterExpiryDate1 xsi:type="xsd:string"><value></remitterExpiryDate1>
<remitterIdType2 xsi:type="xsd:string"><value></remitterIdType2>
```

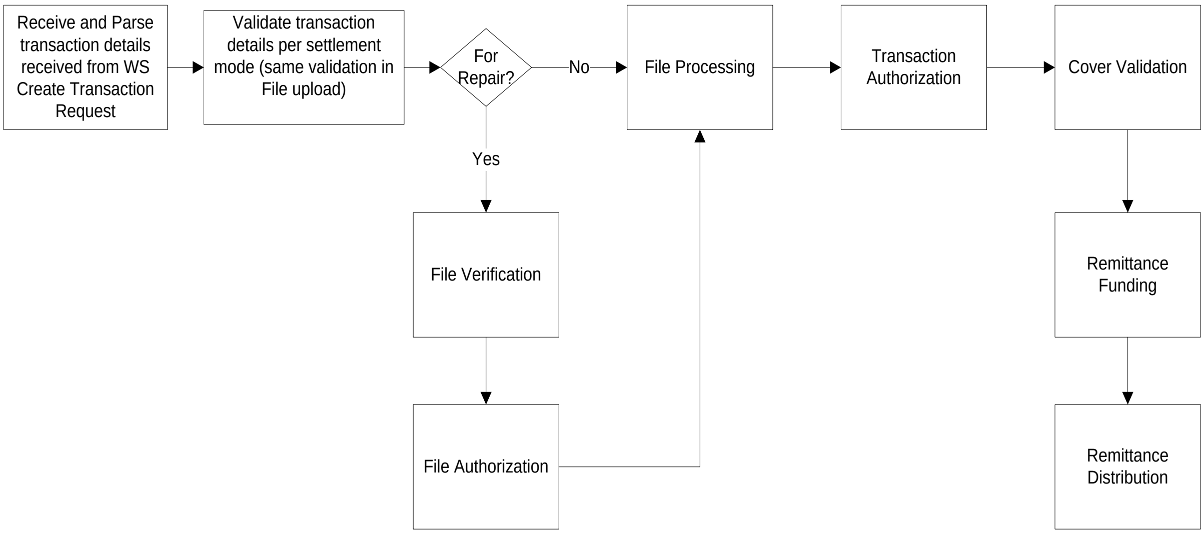
```
<remitterIdNumber2 xsi:type="xsd:string"><value></remitterIdNumber2>
<remitterIdIssueAt2 xsi:type="xsd:string"><value></remitterIdIssueAt2>
<remitterExpiryDate2 xsi:type="xsd:string"><value></remitterExpiryDate2>
<remitterNotificationType><value></remitterNotificationType>
<remitterZipCode><value></remitterZipCode>
<remitterBdate xsi:type="xsd:string"><value></remitterBdate>
<remitterNationality xsi:type="xsd:string"><value></remitterNationality>
<remitterGender xsi:type="xsd:string"><value></remitterGender>
<remitterCivilStatus xsi:type="xsd:string"><value></remitterCivilStatus>
<remitterMobileNumber xsi:type="xsd:string"><value></remitterMobileNumber>
<remitterTin><value></remitterTin>
<remitterContinent xsi:type="xsd:string"><value></remitterContinent>
<remitterOfficePhoneNumber xsi:type="xsd:string"><value></remitterOfficePhoneNumber>
<beneId xsi:type="xsd:string"><value></beneId>
<beneLastName xsi:type="xsd:string"><value></beneLastName>
<benefirstName xsi:type="xsd:string"><value></benefirstName>
<beneMiddleName xsi:type="xsd:string"><value></beneMiddleName>
<beneCustType xsi:type="xsd:string"><value></beneCustType>
<beneAddr1 xsi:type="xsd:string"><value></beneAddr1>
<beneAddr2 xsi:type="xsd:string"><value></beneAddr2>
<beneAddr3 xsi:type="xsd:string"><value></beneAddr3>
<beneCountry xsi:type="xsd:string"><value></beneCountry>
<beneLandmark xsi:type="xsd:string"><value></beneLandmark>
<beneProfession xsi:type="xsd:string"><value></beneProfession>
<benePoBox xsi:type="xsd:string"><value></benePoBox>
<beneOfficePhoneNumber xsi:type="xsd:string"><value></beneOfficePhoneNumber>
<beneEmail xsi:type="xsd:string"><value></beneEmail>
<beneNotificationType xsi:type="xsd:string"><value></beneNotificationType>
<beneZipCode xsi:type="xsd:string"><value></beneZipCode>
<beneBdate xsi:type="xsd:string"><value></beneBdate>
<beneNationality xsi:type="xsd:string"><value></beneNationality>
<beneGender xsi:type="xsd:string"><value></beneGender>
<beneCivilStatus xsi:type="xsd:string"><value></beneCivilStatus>
<beneMobileNo xsi:type="xsd:string"><value></beneMobileNo>
<beneTin xsi:type="xsd:string"><value></beneTin>
<beneRelationToRemitter xsi:type="xsd:string"><value></beneRelationToRemitter>
<altRecRelToBene xsi:type="xsd:string"><value></altRecRelToBene>
<applicationNumber xsi:type="xsd:string"><value></applicationNumber>
<serviceCode><value></serviceCode>
<accountNumber xsi:type="xsd:string"><value></accountNumber>
<bankCode xsi:type="xsd:string"><value></bankCode>
<branchCode xsi:type="xsd:string"><value></branchCode>
<accountType><value></accountType>
<accountName xsi:type="xsd:string"><value></accountName>
<branchName xsi:type="xsd:string"><value></branchName>
<xmitCcyCd><value></xmitCcyCd>
<xmitAmount><value></xmitAmount>
<counterCharge><value></counterCharge>
<isWaived><value></isWaived>
<otherCharges><value></otherCharges>
<beneCcyCd xsi:type="xsd:string"><value></beneCcyCd>
<beneAmtConv><value></beneAmtConv>
<outletCd xsi:type="xsd:string"><value></outletCd>
<bankName xsi:type="xsd:string"><value></bankName>
<cardNo xsi:type="xsd:string"><value></cardNo>
<paymentField1 xsi:type="xsd:string"><value></paymentField1>
<paymentField2 xsi:type="xsd:string"><value></paymentField2>
<paymentField3 xsi:type="xsd:string"><value></paymentField3>
<rtBuyingApplied><value></rtBuyingApplied>
<paymentField4 xsi:type="xsd:string"><value></paymentField4>
<paymentField5 xsi:type="xsd:string"><value></paymentField5>
<outletBranchCd xsi:type="xsd:string"><value></outletBranchCd>
<altBeneName xsi:type="xsd:string"><value></altBeneName>
```

```
<altBeneRelToBene xsi:type="xsd:string"><value></altBeneRelToBene>
<messageToBene xsi:type="xsd:string"><value></messageToBene>
<receiverCorrBank xsi:type="xsd:string"><value></receiverCorrBank>
<senderCorrBank xsi:type="xsd:string"><value></senderCorrBank>
<sendingBank xsi:type="xsd:string"><value></sendingBank>
<receivingBank xsi:type="xsd:string"><value></receivingBank>
<modeOfCharge xsi:type="xsd:string"><value></modeOfCharge>
<purposeCode xsi:type="xsd:string"><value></purposeCode>
<individualCode xsi:type="xsd:string"><value></individualCode>
</arg1>
</com:createTransaction>
</soapenv:Body>
</soapenv:Envelope>
```

Sample Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Body soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<ns1:createTransactionResponse xmlns:ns1="http://com.fowebservice.services.soap">
<result xsi:type="xsd:string"><![CDATA[<response>
<iserror>1</iserror>
<errorMessage><Error message></errorMessage>
<data>
< noOfTransactions><value></ noOfTransactions>
< totalAmountReceived><value></ totalAmountReceived>
< coverNumber><value></ coverNumber>
</data>
</response>]]></result>
</ns1:createTransactionResponse>
</soapenv:Body>
</soapenv:Envelope>
```

Process Flow:


Receive and Parse transaction details received from WS Create Transaction Request
Validate transaction details per settlement mode (same validation in File upload)
For Repair?
No
Yes
File Processing
Transaction Authorization
Cover Validation
File Verification
File Authorization
Remittance Funding
Remittance Distribution

Transaction Detail Inquiry

This facility allow user to request for transaction status processed in Carnelian (CBC). User can inquire multiple transaction status in single request.

Parameter Name	Type	Size	Parameter Description	
Input Parameters				
Header Section				
Username (Required)	String	10	Username that a client will use to authenticate himself inside carnelian.	MANDATORY
Password (Required)	String	---	An encrypted password that will be sent. Jasypt is used for this purpose.	MANDATORY
Remittance Date From	Date (MM/DD/YYYY)	10	Foreign Office date Note: should be mandatory only if user encode the feedback status	MANDATORY only if user also encodes the feedback status
Remittance Date To	Date (MM/DD/YYYY)	10	Foreign Office date Note: should be mandatory only if user encode the feedback status	MANDATORY only if user also encodes the feedback status
Feedback Status	String	10	Valid status: Pending, Positive, Negative, Refunded	OPTIONAL
Cover Number	String	32	System generated cover number for batch of transactions received from WS	OPTIONAL
Application Number	String	40	Unique Application number of the foreign office Note: should be mandatory if Remittance Date From and Remittance Date To are blank	MANDATORY if Remittance Date From and Remittance Date To are blank
Output Parameters				
Header Section				
Is Error	Integer		0 – no error occurred. 1 – Error occurred.	
Error Message	String		An error message in case an error has occurred. Refer to annex B for the complete list of error codes	
Detail Section				
Transaction Status	String		Transaction Step Status Refer to annex a for details	
Transaction	String		Valid Values: Outstanding, Positive, Negative and	

Parameter Name	Type	Size	Parameter Description
Feedback Status			Refunded
Paid date/Refund Date	Date (MM/DD/YYYY)		Applicable if status is POSITIVE (CTA, Bills Payment, OTRC) or CLAIMED/DELIVERED(Branch Pickup, Outlet, Pay Anywhere, Courier/Door to Door, CTOB)
Paid Time/Refund Time	Time (HH:MM)		Applicable if status is POSITIVE (CTA, Bills Payment, OTRC) or CLAIMED/DELIVERED(Branch Pickup, Outlet, Pay Anywhere, Courier/Door to Door, CTOB)
Processed Date	Date (MM/DD/YYYY)		Server Date – Date when the transaction process
Processed Time	Time (HH:MM)		Server Time – Transaction time processed by the system

Note:

Should display Transaction Status, Paid Date/Refund Date and Paid Time/Refund Time if IS_ERROR = 0

Sample Request:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://remittance.webservice.carnelian.ccti.com/">
<soapenv:Header/>
<soapenv:Body>
<com:inquireTransaction soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<arg0>
<username xsi:type="xsd:string"><value></username>
<password xsi:type="xsd:string"><value></password>
<dtFrom xsi:type="xsd:string"><value></dtFrom>
<dtTo xsi:type="xsd:string"><value></dtTo>
<feedbackStatus xsi:type="xsd:string"><value></feedbackStatus>
<coverNo xsi:type="xsd:string"><value></coverNo>
<applicationNo xsi:type="xsd:string"><value></applicationNo>
</arg0>
</com:inquireTransaction>
</soapenv:Body>
</soapenv:Envelope>
```

Sample Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Body soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<ns1: inquireTransactionResponse xmlns:ns1=" http://remittance.webservice.carnelian.ccti.com/">
<result xsi:type="xsd:string"><![CDATA[<response>
<iserror>1</iserror>
<errorMessage><Error message></errorMessage>
<data>
**********record 1**************
<item>
<transactionStatus xsi:type="xsd:string"><value></transactionStatus>
<transactionFeedbackStatus xsi:type="xsd:string"><value></transactionFeedbackStatus>
<dtPaidRefund xsi:type="xsd:string"><value></dtPaidRefund>
<timePaidRefund xsi:type="xsd:string"><value></timePaidRefund>
<dtProcessed xsi:type="xsd:string"><value></dtProcessed>
<timeProcessed xsi:type="xsd:string"><value></timeProcessed>
</item>
**********record 2**************
<item>
<transactionStatus xsi:type="xsd:string"><value></transactionStatus>
<transactionFeedbackStatus xsi:type="xsd:string"><value></transactionFeedbackStatus>
<dtPaidRefund xsi:type="xsd:string"><value></dtPaidRefund>
<timePaidRefund xsi:type="xsd:string"><value></timePaidRefund>
<dtProcessed xsi:type="xsd:string"><value></dtProcessed>
<timeProcessed xsi:type="xsd:string"><value></timeProcessed>
</item>
.
.
</data>
</response>]]></result>
</ns1: inquireTransactionResponse>
</soapenv:Body>
</soapenv:Envelope>
```

ANNEX A

Transaction Step Status	Description displayed in Remittance Inquiry
TAGGED_AS_VALID	Valid Item Pending Cover Validation
TAGGED_AS_INVALID	Invalid Item Pending Cover Validation
FOR_COURIER_REASSIGNMENT	For Courier Reassignment
TAGGED_AS_VALID_FOR_FUNDING_COURIER	Courier Reassignment Item Pending Cover Amend Authorization
TAGGED_AS_INVALID_FOR_FUNDING_COURIER	Courier Reassignment Invalid Item Pending Cover Amend Authorization
TAGGED_AS_DISTRIBUTED	Tagged as Distributed
TAGGED_AS_POSITIVE	Tagged as Positive
TAGGED_AS_FOR_REDELIVERY	Tagged as For Redelivery
TAGGED_AS_UNDELIVERED	Tagged as Undelivered
TAGGED_AS_REROUTED	Tagged as Rerouted
TAGGED_AS_GRANTED_EXTENSION	Granted Extension
NEGATIVE_FEEDBACK_FRWRD_CENTER	Amend Remittance
NEGATIVE_FEEDBACK_REASSIGN_COURIER	Tagged as Distributed (Courier Reassignment)
FRWRD_CENTER_CHANGE_BENE_DETAILS	Item Changed Beneficiary Details Pending Authorization
FRWRD_CENTER_CHANGE_BENE_SRV_DETAILS	Item Changed Service and Beneficiary Details Pending Authorization
FRWRD_CENTER_REFUND_CENTER	Item Tagged as for Refund Pending Authorization
BILLING_STATEMENT_BILLED	Positive Item Awaiting Proof
POSITIVE_FEEDBACK_VERIFIED	Positive Item With Proof
POSITIVE_FEEDBACK_UNVERIFIED	Tagged as Unverified
COURIER_BILLING_BILLED	Courier Has Been Billed
TAGGED_AS_FOR_PROCESS_REQUEST	Pending Approval of Extension Request
FOR_AUTH_NEG_FDBK_FRWRD_CENTER	For Amendment Pending Negative Feedback Authorization
FOR_AUTH_NEG_FDBK_REASSIGN_COUR	For Courier Reassignment Pending Negative Feedback Authorization
AUTH_AMEND_REFUND_CENTER	Authorize Amend Remittance - Refund Remittance
AUTH_AMEND_CHANGE_BENE_DETAILS	Authorize Amend Remittance - Change Beneficiary Details
AUTH_AMEND_CHANGE_BENE_SRV_DETAILS	Authorize Amend Remittance - Change Beneficiary and Service Details
VALID_FOR_FUNDING	Valid Item Pending Funding
INVALID_FOR_FUNDING	Invalid Item Pending Funding
FUNDED_VALID	Funded Valid Item
FUNDED_INVALID	Funded Invalid Item
MISMATCHED_ITEMS	Mismatched Item
TAGGED_AS_VALID_FOR_FUNDING_MISMATCH	Mismatched Item (Valid) Pending Cover Amend Authorization

TAGGED_AS_INVALID_FOR_FUNDING_MISMATCH	Mistmatched Item (Invalid) Pending Cover Amend Authorization
NEWLY_CREATED_REMITTANCE	Newly Created Remittance
TRANSACTION_AUTHORIZATION	Transaction Authorization
UPDATE_ACCOUNT_OPENING	Update Account Opening
AUTHORIZE_ACCOUNT_OPENING	Authorize Account Opening
CREDIT_ACCOUNT_OPENING	Credit Account Opening
UPDATE_CARD_OPENING	Update Card Opening
CREDIT_CARD_OPENING	Credit Card Opening
AUTHORIZE_CARD_OPENING	Authorize Card Opening
REFUNDED	Refunded
OUTSTANDING_REFUNDED	Refunded (Outstanding)
OUTLET_BILLING_BILLED	Outlet Has Been Billed
INSTITUTION_BILLING_BILLED	Institution Has Been Billed
FOREIGN_OFFICE_BILLING_BILLED	Foreign Office Has Been Billed

ANNEX B

Error Messages

CODE	WEB SERVICE ERROR MESSAGE
099	No Errors
001	No Records Found
002	Invalid Username or Password
003	Application Error Occurred NOTE: Code "003" will display if the error cannot be trapped at the application level or those that are technical in nature (such as database is down, communication is down etc.)
004	The user is inactive
005	User is locked out
006	User does not exist
007	User profile has expired
008	User password has expired
009	Community Code does not exist
010	Exceeded the number of maximum allowable transaction
011	Invalid Foreign Office
012	Invalid Funding/Settlement Amount/Currency combination
013	All transactions in the file are duplicates
014	With Duplicate Application Number
015	No service code defined for the specified foreign office
016	No service code mapped
017	Invalid Account Number
018	Bank Name is inconsistent with Bank Code
019	Bank Derived is not an owned bank
020	No Owned Bank derived
021	Bank Derived is not a local bank
022	Bank Derived is not an other bank

CODE	WEB SERVICE ERROR MESSAGE
023	No Other Bank Derived
024	Transaction has no bank code and bank name
025	Invalid Account Type
026	Card Number is inconsistent with the Account Type
027	Invalid Area Code
028	No courier is servicing the area
029	Invalid Outlet Branch Code
030	Invalid Outlet Code
031	Biller account validation error
032	Invalid Institution Code
033	Invalid Settlement Amount
034	Buying Rate is invalid
035	Invalid Funding Amount
036	Invalid Funding Currency
037	Invalid Settlement Currency
038	Please set-up the currency codes for Euro, Philippine Peso and US Dollar using the Update System Parameters function.
039	Invalid Account Number Format:
040	Invalid Account Currency:
041	Invalid Card Number Format
042	The Foreign Office is unauthorized to access the web service
101	Remitter Last Name is required
102	Beneficiary Last Name is required
103	Beneficiary Address 1 is required
104	Application Number is required
105	Funding Currency Code is required
106	Settlement Currency Code is required
107	Remitter Name is required

CODE	WEB SERVICE ERROR MESSAGE
108	Beneficiary Name is required
109	Account Number is required
110	Account Type is required
111	Card Number is required
112	Area Code is required
113	Outlet Code is require
114	Bills Payment Field 1 is required
115	Institution Code is required
116	Username is required
117	Password is required
118	Remittance Date is required if Feedback Status is provided
119	At least one parameter should be provided
201	Remittance Date has exceeded the maximum length
202	Remitter ID has exceeded the maximum length
203	Remitter Last Name has exceeded the maximum length
204	Remitter First Name has exceeded the maximum length
205	Remitter Middle Name has exceeded the maximum length
206	Remitter Customer Type has exceeded the maximum length
207	Remitter Address 1 has exceeded the maximum length
208	Remitter Address 2 has exceeded the maximum length
209	Remitter Address 3 has exceeded the maximum length
210	Remitter Country has exceeded the maximum length
211	Remitter PO Box has exceeded the maximum length
212	Remitter Office Phone Number has exceeded the maximum length
213	Remitter Email Address has exceeded the maximum length
214	Remitter ID Type 1 has exceeded the maximum length
215	Remitter ID Number 1 has exceeded the maximum length
216	Remitter ID Issue At 1 has exceeded the maximum length

CODE	WEB SERVICE ERROR MESSAGE
217	Remitter Expiry Date 1 has exceeded the maximum length
218	Remitter ID Type 2 has exceeded the maximum length
219	Remitter ID Number 2 has exceeded the maximum length
220	Remitter ID Issue At 2 has exceeded the maximum length
221	Remitter Expiry Date 2 has exceeded the maximum length
222	Remitter Notification Type has exceeded the maximum length
223	Remitter Zip Code has exceeded the maximum length
224	Remitter Birth Date has exceeded the maximum length
225	Remitter Nationality has exceeded the maximum length
226	Remitter Gender has exceeded the maximum length
227	Remitter Civil Status has exceeded the maximum length
228	Remitter Mobile Number has exceeded the maximum length
229	Remitter TIN has exceeded the maximum length
230	Remitter Continent has exceeded the maximum length
231	Beneficiary ID has exceeded the maximum length
232	Beneficiary Last Name has exceeded the maximum length
233	Beneficiary First Name has exceeded the maximum length
234	Beneficiary Middle Name has exceeded the maximum length
235	Beneficiary Customer Type has exceeded the maximum length
236	Beneficiary Address 1 has exceeded the maximum length
237	Beneficiary Address 2 has exceeded the maximum length
238	Beneficiary Address 3 has exceeded the maximum length
239	Beneficiary Country has exceeded the maximum length
240	Beneficiary Landmark has exceeded the maximum length
241	Beneficiary Profession has exceeded the maximum length
242	Beneficiary PO Box has exceeded the maximum length
243	Beneficiary Office Phone Number has exceeded the maximum length
244	Beneficiary Email has exceeded the maximum length

CODE	WEB SERVICE ERROR MESSAGE
245	Beneficiary Notification Type has exceeded the maximum length
246	Beneficiary Zip Code has exceeded the maximum length
247	Beneficiary Birth Date has exceeded the maximum length
248	Beneficiary Nationality has exceeded the maximum length
249	Beneficiary Gender has exceeded the maximum length
250	Beneficiary Civil Status has exceeded the maximum length
251	Beneficiary Mobile Number has exceeded the maximum length
252	Beneficiary TIN has exceeded the maximum length
253	Beneficiary Relation to Remitter has exceeded the maximum length
254	Alternate Recipient Relation to Beneficiary has exceeded the maximum length
255	Application Number has exceeded the maximum length
256	Service Code has exceeded the maximum length
257	Account Number has exceeded the maximum length
258	Bank Code has exceeded the maximum length
259	Branch Code has exceeded the maximum length
260	Account Type has exceeded the maximum length
261	Account Name has exceeded the maximum length
262	Branch Name has exceeded the maximum length
263	Funding Currency Code has exceeded the maximum length
264	Is Waive Indicator has exceeded the maximum length
265	Beneficiary Currency Code has exceeded the maximum length
266	Outlet Code has exceeded the maximum length
267	Bank Name has exceeded the maximum length
268	Card Number has exceeded the maximum length
269	Payment Field 1 has exceeded the maximum length
270	Payment Field 2 has exceeded the maximum length
271	Payment Field 3 has exceeded the maximum length
272	Payment Field 4 has exceeded the maximum length

CODE	WEB SERVICE ERROR MESSAGE
273	Payment Field 5 has exceeded the maximum length
274	Outlet Branch Code has exceeded the maximum length
275	Alternate Beneficiary Name has exceeded the maximum length
276	Alternate Beneficiary Relation to Beneficiary has exceeded the maximum length
277	Message to Beneficiary has exceeded the maximum length
278	Receiver Correspondent Bank has exceeded the maximum length
279	Sender Correspondent Bank has exceeded the maximum length
280	Sending Bank has exceeded the maximum length
281	Receiving Bank has exceeded the maximum length
282	Mode of Charge has exceeded the maximum length
283	Purpose Code has exceeded the maximum length
284	Individual Code has exceeded the maximum length