USSD HTTP API SPECIFICATION

Version 1.0

Teletalk Bangladesh Limited

Latest version of this document can be obtained from:
http://www.nixtecsys.com/ussd/ugw-teletalk-http-api.pdf

This page has been left blank intentionally

Table of Contents

Scope of the document:

This document specifies HTTP API provided by USSD Gateway for communicating with third party application.

Information on USSD Gateway Core:

- Vendor (Developer): Nixtec Systems (Bangladesh)
- Connection to Core Network: SIGTRAN (M3UA)
- Capacity: With single MAP license it can handle total 65536 active unique (MO and/or MT) sessions. With more license (say 'n') it can handle 'n' times of 65536 sessions in single hardware.
- TPS: Gateway core is capable to handle 4500+ TPS in current hardware (Single Quad core CPU with moderate RAM) [with more powerful hardware it can be more]
- Development Language: C
- Operating System: Linux
- HTTP API was developed in PHP with MySQL database backend

Requirements:

- Third party client must have to be connected to Teletalk network to make communications with USSD Gateway. Usually a VPN connection through some ISP (Metronet, Telnet, etc.) will serve the purpose. Please contact with your Network Solutions Provider regarding connectivity.
- Third party client should have a working HTTP Server to receive USSD sessions initiated by Mobile Users.
- For initiating USSD Push (MT) targeted to Mobile Users from a third party application client will call a HTTP URL with properly filled up parameters given by Teletalk.
- Any HTTP URL call should produce some 'Response Header' and 'Response Body' according to the specification in this document.
- USSD Gateway will NOT store any 3rd party transaction states (but may store CDR for billing/reporting purpose). Third party clients will be responsible to develop their applications based on the specification in this document.

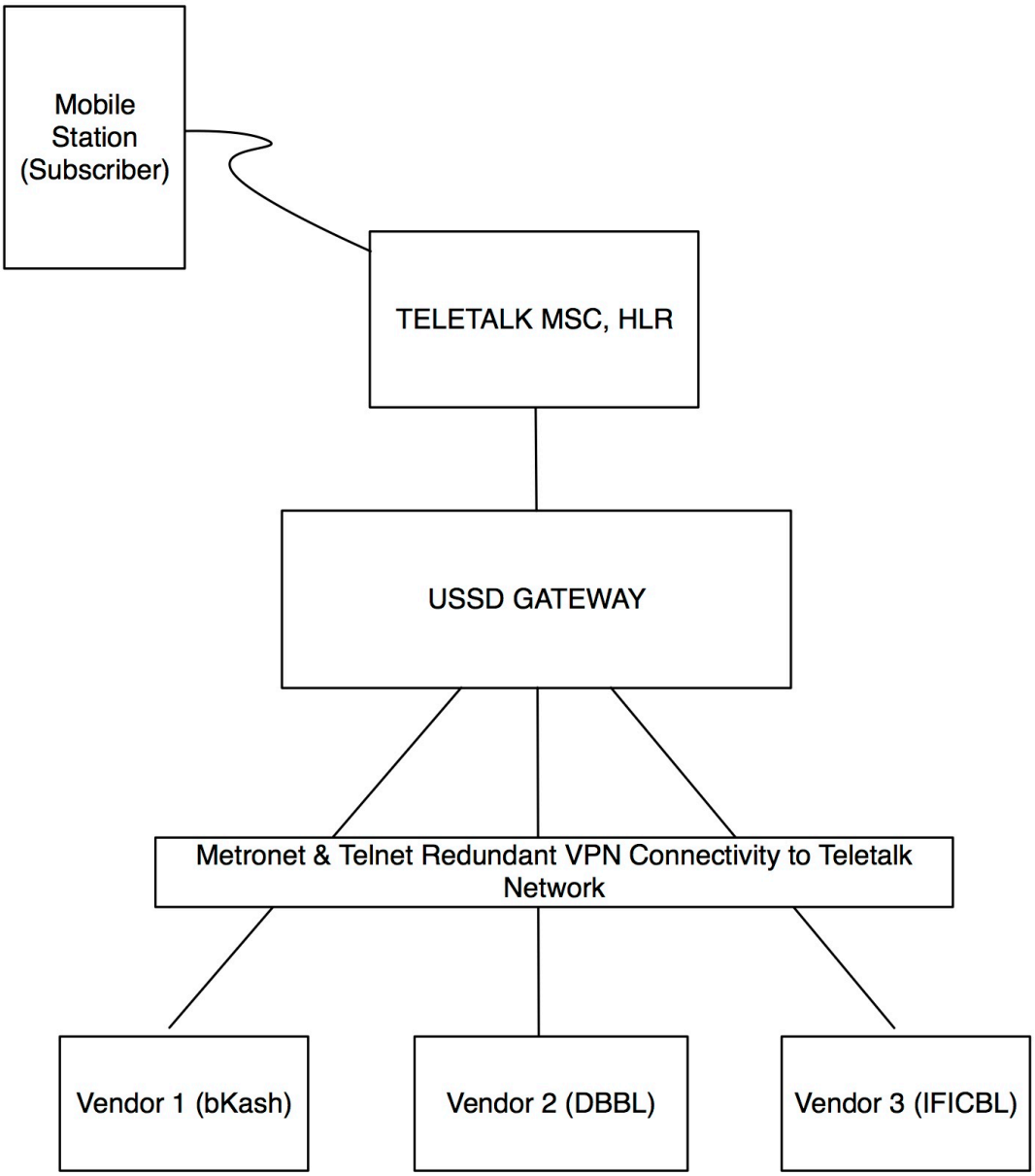

Fig: USSD Gateway Integration Modality

Type of USSD Sessions:

- USSD Sessions (Transactions) can be created two ways:
  - Mobile User Initiated (e.g., when user starts the session by dialing *shortcode#) (Mobile Originated) [MO]
  - Application Initiated (e.g., when 3rd party will start the session to Mobile User) (Mobile Terminated) [MT]

HTTP API for USSD Sessions (Transactions) initiated by a Mobile User [MO]:

- Any user input, as well as the initiating dial of USSD shortcode will be sent to third party URL using GET/POST (as suggested by third party) HTTP method.
- Each USSD Session (Transaction) will be assigned a unique Transaction ID. The ID will remain same from start to end of the USSD session (Transaction). The ID will be passed to third party using 'tid' parameter as in the following table (Table-01).
- A USSD Session (Transaction) will have three states: 1. Begin, 2. Continue, 3. End. The state of the session will be passed using 'flow' parameter as mentioned in following table (Table-01). This document considers a USSD session from 'begin' to 'end' with zero or more 'continue' states and hence it is considered 'Transaction' similar to the concept of database transaction.
- Mobile User's MSISDN (Mobile Number) will be passed to third party using 'msisdn' parameter as in the following table (Table-01).
- Mobile User's input (including the number input at the start of dial) will be passed using 'text' parameter as in the following table (Table-01).
- Following parameters will be sent in each user input (including initial dial).

Table-01

Parameter Name	Description	Type of Value	Length of Value	Example Value
tid	Transaction ID (Unique among running Transactions or Sessions)	Numeric (base 10)	Up to 10 digits. Configurable to add prefix '0's as padding to make it	65535 0000065535

			fixed length if required by 3rd party	
msisdn	Mobile Number of User	Numeric Digits	13 digits	8801550155081
text	User Input	ASCII Characters (7bit GSM)	160 octets* (182 7-bit chars max)	*515# abcd 1234
flow	State of Transaction	Alphabetic (English)	3-8 chars	begin continue end
momt	Mobile Originated (MO) or Mobile Terminated (MT/Push)	Alphabetic	2 chars	mo mt (in case of Push)
user	Username given for teletalk (for security purpose) by 3rd party	Alphanumeric	3-8 chars	t3letalk teletalk teleuser
pass	Password assigned to the 'user' configured above	Alphanumeric	4-8 chars	t123p455

- Knowing the above parameters 3rd party will prepare their URL and share with Teletalk accordingly.
- The third party application will provide output to 'Response Body', which will be sent to user as it is. Currently 7-bit GSM alphabets are supported.
- If the application wants to carry on with user input then it would set Response HTTP header 'X-Flow' to 'continue'. This will tell the USSD Gateway to continue session by allowing user to provide input. In this way, the user will get the text (written in response body) and will be able to enter some input to continue the session with application.
- When the application wants to end session with user it will set HTTP header 'X-Flow' to 'end' which will send the application's output in

response body to user and end the session with user (user will not get any chance to enter anything to continue further).

- If the application doesn't set the 'X-Flow' Response HTTP header to 'continue', USSD GW HTTP API will consider this 'end' by default and end session with user.
- At any time of the running USSD session (transaction) the mobile user can end the session by 'Cancel' or 'End' button in mobile without giving an input, even when the 3rd party was willing to have a reply. In such cases USSD gateway will call 3rd party URL with 'flow' set to 'end' and 'text' (will be empty) should be ignored (as user terminated without giving input).
- If USSD gateway detects interruption of the session (transaction) due to network or some unexpected reasons, it will generate an "end" of the session (transaction) and notify in 3rd party URL as the previous clause.
- For whatever reasons, if USSD Gateway sends "end" to 3rd party URL, the response HTTP header and response body from 3rd party will be discarded (since there will be no way to continue the USSD session / transaction) and gateway doesn't store any 3rd party information exchanged in the lifetime of a session.
- For some reason, if an "end" is lost due to network problem or some other reasons, and the transaction ID (tid) repeats with a 'begin' (with a possibly different MSISDN), 3rd party must consider it new session. It's the responsibility of the 3rd party to make necessary cleanup in their application for the 'begin' with a transaction ID (tid) used in previously processed USSD session.
- It will be guaranteed by USSD Gateway to not to have two same transaction IDs for two 'running' USSD sessions.

Examples of API:

1. Mobile User dials *515#
2. Gateway invokes the 3rd Party URL with given parameters with 'flow' set to 'begin'
3. 3rd party application processes the information and replies with some output for user. It further sets 'X-Flow' HTTP header to 'continue' if the user needs to provide some input. Otherwise it sets the HTTP header 'X-Flow' to 'end'.
4. If user provides no input and cancels/quits the USSD session, Gateway will invoke the URL with 'flow' set to 'end' and no further USSD communication on the same session is possible.
5. If user provides input it is sent to application in 'text' parameter and 'flow' is set to 'continue'.

Example 3rd Party application:

Following is a sample PHP code for communicating with HTTP GET/POST API. 3rd party will get some basic idea on how they will communicate with USSD Gateway through HTTP API. For simplicity 'user' and 'pass' parameters are not mentioned (since it is not, by any means, part of the USSD session). For other programming languages it would work in the similar logic:

The code will keep taking input from the user until he/she puts a '0' as input or ends the call by pressing 'End' button in mobile.

For example, third party will save the code as 'ussd.php' and put to some place in their HTTP Server (Web Server) and share the URL with Teletalk (e.g., : http://192.168.99.99/teletalk/ussd.php). Here 192.168.99.99 is IP address of the HTTP Server that is reachable from Teletalk USSD Gateway.

```php
<?php

# Get Transaction ID for USSD Session
$tid = $_REQUEST['tid'];
# MSISDN of the user
$msisdn = $_REQUEST['msisdn'];
# Number dialed or Input from user
$text = $_REQUEST['text'];
# Session Status {begin,continue,end}
$flow = $_REQUEST['flow'];

# Application Logic will work here
if ($text == "0") {
  $flow = "end";
} else {
  $flow = "continue";
}
# Set header, so that Gateway knows that application wants to carry on the
session with user
header("X-Flow: $flow");

if ($text == "0") {
  echo "Thank You.";
} else {
  echo "USSD Demo\nText: $text\nEnter something to continue, 0 to quit";
}

die();

?>
```

HTTP API for USSD Sessions (Transactions) initiated by a 3rd Party Application [MT]:

The MT/Push USSD Sessions are initiated by 3rd party the same way USSD Gateway does a 'begin' of session. Here Teletalk will give a URL for 3rd party which they will call the same way as MO data sent to third party. One exception is that the 'tid' (Transaction ID) field will be generated by USSD gateway as 'Response Body' instead of clients passing it in the URL. For clarification the following table (Table-02) shows the list of parameters to be passed to the MT URL:

Table-02

Parameter Name	Description	Type of Value	Length of Value	Example Value
msisdn	Mobile Number of User	Numeric Digits	13 digits	8801550155081
text	Text data to send to mobile user to initiate USSD MT session	ASCII Characters (7bit GSM)	160 chars* (182 7-bit chars max)	Welcome to Mobile Banking: 1. Check Balance 2. Send Money 0. Exit
flow	State of Transaction	Alphabetic (English)	3-8 chars	begin end
user	Username given by teletalk (for security purpose)	Alphanumeric	3-15 chars	3rdparty
pass	Password assigned to the 'user' mentioned above	Alphanumeric	4-15 chars	3rdpartypass

- The HTTP URL call issued by 3rd party to Teletalk will produce Response Body that will contain the transaction id (tid) for the session being initiated. Thus it can be considered that the Teletalk's URL invocation is just to get the Transaction ID (tid) to begin a USSD session. Subsequent

user inputs will be transferred to 3rd party the same way as MO USSD session.

- When there will be user input (in case client set 'flow' to 'begin'), the 3rd party will be notified in the same way like MO USSD sessions with those parameters in the MO USSD API. Just in this case 'momt' parameter will have 'mt' value.
- If third party wants just to notify the Mobile User with some message and no user input is expected, the value of 'flow' parameter in MT USSD invocation should be set to 'end'.

Basic differences between MO and MT USSD sessions:

Table-03

Operation	USSD MO Session	USSD MT Session
1. Initiation	Initiated by Mobile User	Initiated by 3rd party application
2. URL Invocation to 'begin' transaction	Teletalk invokes 3rd party URL with given parameters	3rd party application invokes Teletalk URL with given parameters
3. momt Parameter	Value of 'momt' parameter will be always 'mo'	Value of 'momt' parameter will be always 'mt'
4. Session Cleanup	Third party should clean session when they get a 'flow=end' from gateway or when they send the 'X-Flow' header set to 'end'	Third party should clean session when they get a 'flow=end' from gateway. Because USSD Gateway will acknowledge the application when the user gets the final text.

Possible Error cases for MT USSD API Call:

In case of successful calling of a USST MT request, Gateway will reply with a numeric transaction ID in HTTP 200 response body, which will be used by application for further communications on the USSD session. Error cases in calling a MT session request are as follows:

Serial	Error Response from API	Error Description
1	ERROR [PARAMETER]	Required parameter missing
2	ERROR [AUTH]	Username and/or password not matched

3	ERROR [MSISDN]	Invalid MSISDN passed
4	ERROR [TEXT]	Empty text passed
5	ERROR [TEMPORARY]	Temporary error from gateway, try later

NB: All error messages start with 'ERROR'.

Note on Length of Text:

In GSM 09.02 (MAP) 160 octets is stated as the maximum length for the USSD string. Due to underlying signaling layers the maximum length of the USSD string depending on the message is as following.

Table-04

USSD operation	Max length
Begin, Invoke ProcessUSSDRequest	133
End, Result ProcessUSSDRequest	160
First Continue, Invoke USSDRequest in mobile initiated dialogue	154
Begin, Invoke USSDRequest	144
First Continue, Result USSDRequest in network initiated dialogue	154
Other messages	160

If we consider 7-bit GSM Alphabet (that is used in most handsets) it can fit in 7-bit. Hence maximum 182 ((160x8)/7) such characters can be transferred using USSD. System currently supports only 7-bit GSM Alphabet. It may support Unicode in near future.

Various timeouts:

- A whole running session continues as long as Core Mobile Network holds it with subscriber.
- USSD Gateway doesn't by-itself imposes any time limit in taking user input to application.
- USSD Gateway has to detect a time-out from Application side when waiting for a response text to be delivered to subscriber.
- In case the Core Network applies time-out to some session, USSD gateway will receive the signal about the timeout and will clean up accordingly.
- In case USSD Gateway receives a 'Session End' (or aborted) signal from core network, it will perform a cleanup and invoke third party URL with 'text' set to blank (empty) and 'flow' set to 'end'. Third party may perform a cleanup upon receiving an "end", because no more communication is

possible on an ended session. Gateway will ignore any type of response from application when invoking third party URL with 'flow=end'.

- In case there is a timeout in anywhere in communication (Mobile Handset/Core Network/USSD Gateway/3rd Party Application), it should be considered as interruption to that particular session

SL	Description	Timeout by Core Network	Timeout by USSD Gateway
01	Whole lifetime of a session	Configured by Operator (usually no timeout unless there is timeout due to being idle in one dialogue)	No timeout
02	User input to send to network	Configured by Operator (usually 90-120 seconds)	No timeout
03	Response from 3rd party application	Configured by Operator. Also handset may have some time-out mechanism waiting for response from core network	15 seconds defined in the HTTP API by default. If 3rd party needs more time (up to 60) to process a request to make a response they must request operator explicitly to configure it in API.
04	Other sort of timeout	Depends on the type of ongoing transaction	No timeout

Example MO USSD Session in the application point of view:

1. Mobile User (having MSISDN 8801550155081) dials *515#. Gateway invokes (Note that '*' and '#' are hex-encoded, same applies to any character that needs to be encoded in a URL: http://3rdparty.ip/teletalk/ussd.php?user=teletalk&pass=telepass&tid=65535&msisdn=8801550155081&text=%2A515%23&flow=begin&momt=mo
2. 3rd party application responses with Response Header 'X-Flow: continue' and Response Body 'Please enter some input'. Gateway transfers this

message to mobile user. User gets it in his/her mobile screen with option to enter something as input.

3. Mobile user inputs "1234". Gateway invokes: http://3rdparty.ip/teletalk/ussd.php?user=teletalk&pass=telepass&tid=65535&msisdn=8801550155081&text=1234&flow=continue&momt=mo
4. 3rd party application responses with Response Header 'X-Flow: end' and Response Body 'Thank you for your feedback'. Gateway transfers this message to mobile user. User gets it in his/her mobile option but can't provide any input to carry on the session.
5. If Mobile User doesn't provide any input in 3rd point above and ends the session by pressing 'End' button in handset, Gateway invokes: http://3rdparty.ip/teletalk/ussd.php?user=teletalk&pass=telepass&tid=65535&msisdn=8801550155081&text=&flow=end&momt=mo

Example MT USSD Session in the application point of view:

1. 3rd party wants to start a USSD session with user. 3rd party invokes: http://teletalk.ip/ussd/mt.php?user=3rdparty&pass=3rdpartypass&msisdn=8801550155081&text=Please%20Enter%20Something&flow=begin
2. Teletalk generates a transaction ID (tid) (e.g., 65535) for this operation and gives it in the HTTP Response Body. 3rd party will store it to keep track of the USSD Session (transaction) states.
3. Teletalk sends a USSD push to user with the text. Since it was a 'begin' in 'flow' parameter, user will get provision for putting input.
4. When user gives input (e.g., 1234) and click 'Send' in mobile, Teletalk invokes the USSD MO URL accordingly (similar way as MO, except that the 'momt' parameter set to 'mt'): http://3rdparty.ip/teletalk/ussd.php?user=teletalk&pass=telepass&tid=12345678&msisdn=8801550155081&text=1234&flow=continue&momt=mt
5. 3rd party will continue session with user similar way like MO session.

Example MT USSD Notify User in the application point of view:

1. 3rd party wants just to notify user with some message and doesn't expect any input back. 3rd party invokes:

http://teletalk.ip/ussd/mt.php?user=3rdparty&pass=3rdpartypass&msisdn=8801550155081&text=Your%20Month%20end%20balance%20is%201234.50%20Taka.&flow=end

2. Teletalk generates a transaction ID (tid) (e.g., 65535) for this operation and gives it in the HTTP Response Body. 3rd party should store it for cleanup purpose.
3. Teletalk sends a USSD push to user with the text. Since it was an "end" in 'flow' parameter, user will get no provision for putting input.
4. When the notification text reaches user's handset, the USSD Gateway will acknowledge the 3rd party about the receiving by calling URL the same way as a MO USSD session ends if user closes without giving input. Just in this case 'momt' parameter will have value 'mt': http://3rdparty.ip/teletalk/ussd.php?user=teletalk&pass=telepass&tid=65535&msisdn=8801550155081&text=&flow=end&momt=mt.
5. Third party application will clean up the session upon receiving the 'end' flow from gateway.