



# HTTP REST API Interface Specification

[HTTP REST API Interface specification technical manual. Provision of services is dependent upon compliance with the specifications set forth herein. The information in this document is subject to change without notice. Although World Text has taken reasonable steps to ensure the accuracy and completeness of this document, it shall not be liable for any losses whatsoever, whether direct or indirect, including without limitation and loss of profit, loss of use, or loss of data, as a result of any errors or omissions contained herein. The information or statements in this document concerning the specification or performance of World Text software or hardware systems shall not constitute any binding promise or warranty.]

World Text, PO Box 226, Chester le Street, County Durham. DH3 9BR  
Phone (UK): 0845 867 3980 Phone (Int) +4420 704 37037

## Table of Contents

<b>Direct Connection Details</b> .....	<b>3</b>
<b>Proxy Connection Details</b> .....	<b>3</b>
<b>Method Summary</b> .....	<b>3</b>
Authentication Methods .....	3
Delivery & Query Methods.....	3
Group Management Methods .....	3
Miscellaneous Methods.....	4
<b>Parameter Description</b> .....	<b>4</b>
<b>List Parameters</b> .....	<b>4</b>
<b>Return Values</b> .....	<b>4</b>
<b>Authentication</b> .....	<b>5</b>
<b>Username &amp; Password</b> .....	<b>5</b>
<b>Authentication Key</b> .....	<b>5</b>
<b>Method Detail</b> .....	<b>5</b>
credits .....	5
delauthkey .....	5
getauthkey .....	5
groupadd .....	6
groupdel.....	6
groupdelall.....	6
groupget.....	6
groupgetdetails.....	6
querysms .....	6
ping .....	6
sendgroup .....	7
sendsms .....	7
submitsm.....	7
wappush .....	7
<b>Failure Codes</b> .....	<b>7</b>
<b>Message Status Callback</b> .....	<b>8</b>
<b>Callback Parameters</b> .....	<b>8</b>
<b>HTTP Interface with PHP Example</b> .....	<b>9</b>

## Direct Connection Details

For clients that connect to ports other than the HTTP/S ports (80/443) the preferred method of connection is directly to the HTTP/S service using the hostname and port numbers below, for clients that can only connect out on standard ports please see Proxy Connection Details:

	Non SSL	SSL Connections
Hostname	sms.world-text.com	sms.world-text.com
Port	1081	1082

Example call from a browser address bar: <http://sms.world-text.com:1081/ping>

## Proxy Connection Details

For clients that can only connect out on the HTTP/S ports then all calls should be made via the WorldText website proxying API for the HTTP service.

	Non SSL	SSL Connections
Hostname	www.world-text.com	www.world-text.com
Port	80	443
URL Prefix	/proxyapi	/proxyapi

Example call from a browser address bar: <https://www.world-text.com/proxyapi/ping>

## Method Summary

Authentication Methods	
Method	Description
delauthkey	Deletes an allocated authorisation key
getauthkey	Retrieves an authorisation key for further transactions

Delivery & Query Methods	
Method	Description
querysms	Query the delivery status of a message
sendsms	Sends a text message to a mobile device or UK land line
submitsm	Not yet implemented
wappush	Delivers a basic WAP Push message to a mobile device

Group Management Methods	
Method	Description
groupadd	Add a new number to an existing group
groupcreate	Create a new group list, this is a list of numbers that messages can be sent to in one call
groupdel	Deletes a number from an existing group
groupdelall	Deletes all numbers from an existing group

groupget	Retrieves all the numbers from an existing group
groupgetdetails	Retrieves all the numbers and names from an existing group
groupremove	Remove an existing group list
sendgroup	Dispatches a message to a predefined group

Miscellaneous Methods	
Method	Description
credits	Retrieves the amount of credit available on the specified account
ping	Test interface availability & keep session key alive

## Parameter Description

Please note that items in *italics* are in beta test and may be withdrawn or modified at any time.

Name	Type	Max Size	Description
accid	4-6 digit number	6	Account ID
AUTH	key or user/pass	n/a	See Authentication section
callback	URL	n/a	Message Status callback HTTP URL
data	HEX string	unlimited	Encoded message content
dstaddr	number list	25 numbers	Destination Address List (recipients)
groupid	1-6 digit number	6	Group list ID
key	HEX string	32	Authorisation key (see getauthkey)
msgid	HEX string	32	Message ID
multipart	1 digit number	1	Max number of SMS parts (default 1)
name	string	20	A user defined name
none	none	none	No argument is required for this call
pass	string	50	Account Password
srcaddr	string or number	16	Source Address (sender)
txt	ISO 8859 string	unlimited	Plain text string to be sent
url	URL	n/a	Uniform Resource Locator
user	string	50	Account Username

## List Parameters

Parameters that are specified as being a list can consist of one or more values separated by a comma symbol (,). An example would be the *dstaddr* field, following is an example of specifying values for these fields

## Return Values

Every line returned as a result from the HTTP interface is prefixed by the values shown in the table below.

SUCCESS	the request was either fully or partially successful
FAIL	the request failed

Following the above strings will be the method return values, for certain interface calls there may be more than one result line returned.

## Authentication

Most calls to the HTTP/S interface require that authentication parameters are sent along with the request, this can be handle one of two ways:

### Username & Password

For every call to the interface requires that the username and password fields (user & pass) are provided. This method is simple however is less secure on non SSL connections.

### Authentication Key

For every call to the interface an authentication key (key) is passed. This key is obtained by an initial call to `getauthkey` and when the sessions is finished it is removed by a call to `delauthkey`, or the key will expire after 10 minutes of inactivity. This method is the most secure option and for bulk use a quicker means of authentication.

## Method Detail

Please note that items in *italics* are in beta test and may be withdrawn or modified at any time.

### credits

<b>Description</b>	Retrieves the amount of credit available on the specified account
<b>Required Args</b>	AUTH
<b>Return Value</b>	credits remaining on the account, e.g: SUCCESS 999
<b>Policy</b>	Do not check more that once every 120 seconds

### delauthkey

<b>Description</b>	Deletes an allocated session authorisation key
<b>Required Args</b>	key
<b>Return Value</b>	OK on success

### getauthkey

<b>Description</b>	Allocates and retrieves a session authentication key that is valid for one session. The key expires after a predefined period, it is also removed on application restarts and should be deleted when finished with.
<b>Required Args</b>	user, pass
<b>Return Value</b>	HEX string value

## groupadd

<b>Description</b>	Delete an entry from a group
<b>Required Args</b>	AUTH, grpId, dstAddr, name
<b>Return Value</b>	SUCCESS or FAIL

## groupdel

<b>Description</b>	Delete an entry from a group
<b>Required Args</b>	AUTH, grpId, dstAddr
<b>Return Value</b>	SUCCESS or FAIL

## groupdelall

<b>Description</b>	Deletes all entries from a group
<b>Required Args</b>	AUTH, grpId
<b>Return Value</b>	SUCCESS <entries deleted> or FAIL

## groupget

<b>Description</b>	Retrieves all the numbers in a group
<b>Required Args</b>	AUTH, grpId
<b>Return Value</b>	Numbers on consecutive lines or FAIL

## groupgetdetails

<b>Description</b>	Retrieves all the numbers and names in a group
<b>Required Args</b>	AUTH, grpId
<b>Return Value</b>	Numbers and names on consecutive lines or FAIL, e.g.: "0700000000","Alan B Smith"

## querysms

<b>Description</b>	Query the delivery status of a message
<b>Required Args</b>	AUTH, msgId
<b>Return Value</b>	Message status value
<b>Policy</b>	Initial check for delivery at a minimum of 15 seconds after submission. All further checks should be double the previous time frame, in the case the next check would be at 30 seconds, then 60 seconds etc.

## ping

<b>Description</b>	Test interface availability & optionally key session key alive
<b>Required Args</b>	none
<b>Optional Args</b>	key
<b>Return Value</b>	SUCCESS or FAIL

## sendgroup

<b>Description</b>	Sends a text message to a group
<b>Required Args</b>	AUTH, grpId, txt
<b>Optional Args</b>	srcaddr, multipart, test
<b>Return Value</b>	Number of messages dispatched

## sendsms

<b>Description</b>	Sends a text message to a mobile device or UK land line
<b>Required Args</b>	AUTH, dstaddr, txt
<b>Optional Args</b>	srcaddr, multipart, callback
<b>Return Value</b>	Multiline response of a message id followed by the credits remaining (if applicable) on the account, there will be one line for each message dispatched eg: SUCCESS 123 999 SUCCESS 124 989

## submitsm

<b>Description</b>	Not yet implemented
--------------------	---------------------

## wappush

<b>Description</b>	Delivers a basic WAP Push message to a mobile device
<b>Required Args</b>	AUTH, dstaddr, txt, url
<b>Optional Args</b>	srcaddr
<b>Return Value</b>	Multiline response of a message id followed by the credits remaining (if applicable) on the account, there will be one line for each message dispatched eg: SUCCESS 123 999 SUCCESS 124 989
<b>Restrictions</b>	url and txt parameter limited to 154 characters combined

## Failure Codes

If a request does not result in success then the response will be **FAIL code** where *code* is one of the following failure codes.

Failure Code	Description	Application Action
ARGUMENTS	The wrong number of arguments were supplied. Refer to the specification.	Do not retry
AUTHORISATION	The credentials used to authenticate the account were specified incorrectly.	Do not retry
BLACKLISTED	The MSISDN is blacklisted, do not retry,	Do not retry
CREDIT	Not enough credit remaining on the account.	Do not retry
DESTINATION	MSISDN specified was not long enough or in the correct format.	Do not retry

Failure Code	Description	Application Action
DISABLED	Account being used is disabled or suspended, contact support.	Do not retry
EXCEPTION	An application exception occurred, contact support.	Do not retry
INTERNAL	An internal routing error occurred, please retry your message.	Retry immediately
INVALIDARGS	The arguments failed to meet the method requirements.	Do not retry
KEYEXPIRED	The key has expired or does not exist.	Complete authentication
MESSAGEID	Invalid or blank messages ID, contact support.	Do not retry
SOURCEADDR	An unallocated source address was specified.	Do not retry
SUBMITRESP	Incorrect SMPP response received, please retry your message.	Retry immediately
TIMEOUT	SMPP Response not received in time, retry your message.	Retry immediately
THROTTLED	To much traffic is being submitted through this interface.	Retry after 20 seconds
UNKNOWNCMD	Command is unknown	Do not retry
VERSION	Incorrect interface version specified	Do not retry

## Message Status Callback

Optionally an HTTP URL can be specified that is called once when a message status changes, the normal status change is to a final state such as DELIVRD which indicates that the message has been delivered. The call is to a standard HTTP POST to a URL and should be specified in full, e.g. "http://myserver.com/smsstatus.php".

## Callback Parameters

Parameter	Description
error	Error status of the message (see Error Codes in Documentation)
msgid	Message ID that was returned when the submission was made
state	State of the message, DELIVRD, EXPIRED, UNDELIV, ACCEPTD, REJECTD

## HTTP Interface with PHP Example

```
function doPostRequest( $host, $port, $uri, $body ) {
    $url = "http://$host:$port/$uri";
    $ch = curl_init();

    // set the target url
    curl_setopt($ch, CURLOPT_URL,$url);

    // howmany parameter to post
    curl_setopt($ch, CURLOPT_POST, 1);

    // the parameter 'username' with its value 'johndoe'
    curl_setopt($ch, CURLOPT_POSTFIELDS,"$body");

    // Return the data rather than printing
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);

    $result = curl_exec( $ch );
    curl_close( $ch );
    return $result;
}

function sendSMS( $username, $password, $dstaddr,
                 $message, $srcaddr = "World-Text" )
{
    $message = urlencode( $message );
    $result = doPostRequest( "sms.world-text.com", 1081, "sendsms",
    "username=$username&password=$password&sourceaddr=$srcaddr&dstaddr=$dstaddr&
    txt=$message" );
    return $result;
}

sendSMS( "username", "password", "44777111222", "Hello this is a PHP SMS
Test from World-Text using the HTTP Interface", "SMSAlert" );
```