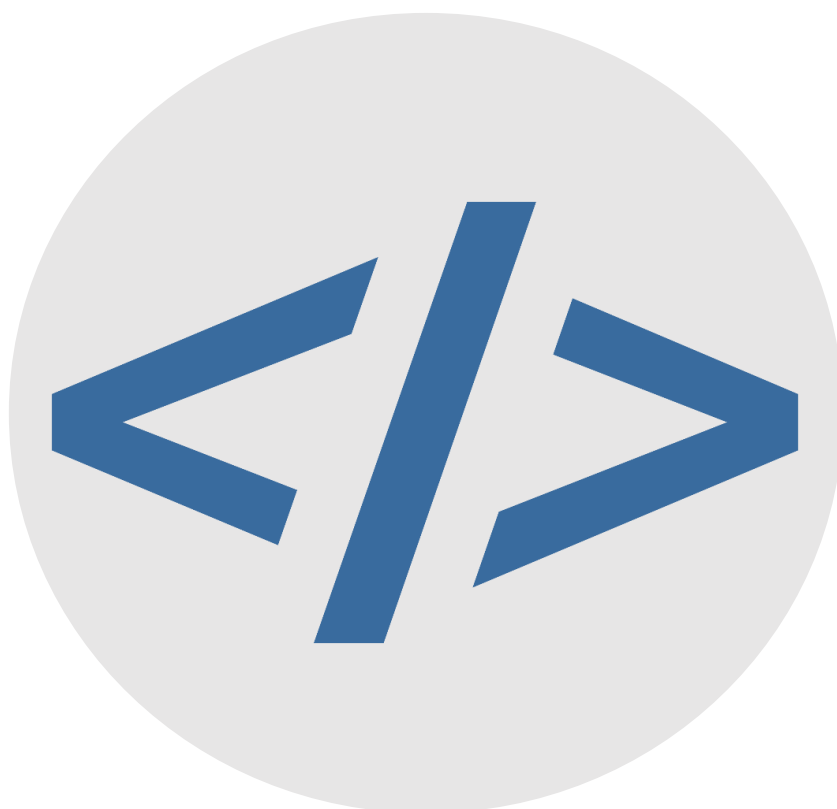




WinSMS API SPECIFICATIONS

UPDATED 23RD MAY 2018



DISCOVER THE EASY WAY TO SEND BULK SMS

www.WinSMS.co.za

Important Notes

Number Format:

All mobile numbers must be specified in international format - EXAMPLE: 083xxxxxxx
MUST BE SENT AS 2783xxxxxxx. (The leading "+" is assumed)

Character Set:

Text content must be restricted to the available character set, as listed in the official GSM 03.38 specifications, which may be downloaded from www.etsi.org

HTTP Request Length:

The length of any single HTTP request to the HTTP API is restricted. Please restrict individual calls to the pages to 100 recipients or less.

URL Encoding:

Please ensure that ASCII character encoding is used BEFORE applying any URL encoding.

All non alphanumeric characters in the message should be URL encoded. However, to avoid problems it is advisable to URL encode the entire message. For more information on URL encoding go to http://www.w3schools.com/TAGS/ref_urlencode.asp

XML Gateway:

ALWAYS remove ALL Carriage Returns from any XML document posted to any of the XML Gateway locations. **To include a carriage return in the 'Message' node please use the URL-encoded value for a line feed: %0A.** The examples in this document include Carriage Returns to make them more legible.

Gateway Polling:

When checking the status of sent messages or getting replies, restrict your polling interval to a minimum of 2 minutes. Polling too often will return "**TOOSOON**".

GSM Encoded Length

The length of a message as it is calculated by all WinSMS end points. ie. 'special' characters count as two characters.

NB: The following single characters count as 2 characters in ALL APIs:

| (pipe)
^
{
}
[
]
~
\

GSM Encoded Segment Count

Probably more important than the length of a message. This is the number of segments (or message parts) needed to send a single SMS message. And has a 1:1 relationship with the credit cost of a message.

ie. An SMS message with a GSM encoded length of 1 to 160 will have a GSM Encoded Segment Count of 1 (1 credit cost to ZA mobile #), an SMS message with a GSM encoded length of 161 to 306 will have a GSM Encoded Segment Count of 2 (2 credit cost to ZA mobile #).....

The WinSMS API supports a maximum of 6 segments (GSM encoded length of 918 characters)

Spaces:

Two or more spaces in a message will be replaced with one space. Any trailing spaces will be removed from the message.

Time Zone:

The Time Zone is GMT+2 - Greenwich Mean Time.

NB: All scheduled Messages will be scheduled in GMT+2.

Document Contents

[XML API](#) [Page 4](#)

[Send Batch](#)
[Batch Status](#)
[Delete Scheduled](#)
[Replies](#)

[HTTP API](#) [Page 21](#)

[Authentication](#)
[Credit Balance](#)
[Send Batch](#)
[Batch Status](#)
[Delete Scheduled](#)
[Replies](#)
[List Opt-Outs](#)
[Credit Transfer](#)

[FTP API](#) [Page 33](#)

[E-mail to SMS](#) [Page 34](#)

[Subject Authentication](#)
[Sender Address Authentication](#)

[Outlook Mobile Service](#) [Page 38](#)

[Outlook Mobile Service for Outlook 2007](#)
[Outlook Mobile Service for Outlook 2010](#)

[Standard Rate Short/Long Code](#) [Page 41](#)

[Requesting Short/Long Code Messages via HTTP](#)
[Requesting Short/Long Code Messages via XML](#)

[Reply and Delivery Report Callbacks](#) [Page 49](#)

WinSMS XML Gateway API

Send Batch

Description:

This command submits one message to a batch of cell numbers. It is also used for sending individual messages.

Current Location: Call: **<http://www.winsms.co.za/api/XMLSendSMSMulti.asp>**

HTTP Post an XML file with the following layout (**Use exact capitalisation**):

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="WinSMS_Multi_Message" type="WinSMS_Multi_MessageType" />
<xs:complexType name="WinSMS_Multi_MessageType">
<xs:annotation>
<xs:appinfo db:docElement="WinSMS_Multi_Message" />
</xs:annotation>
<xs:sequence>
<xs:element name="User" type="xs:string" />
<xs:element name="Password" type="xs:string" />
<xs:element name="Scheduled" type="xs:integer" />
<xs:element name="Messages" type="MessagesType" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="MessagesType">
<xs:sequence>
<xs:element name="Message" type="MessageType" maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="MessageType">
<xs:sequence>
<xs:element name="MessageText" type="xs:string" />
<xs:element name="Numbers" type="NumbersType" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="NumbersType">
<xs:sequence>
<xs:element name="Number" type="xs:integer" maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Example XML:

```
<WinSMS_Multi_Message>
  <User>ianw</User>
  <Password>thegreat</Password>
  <Scheduled>201111141025</Scheduled>
  <Messages>
    <Message>
      <MessageText>Sample Message 1</MessageText>
      <Numbers>
        <Number>27825550101</Number>
        <Number>27835550505</Number>
        <Number>27845550909</Number>
      </Numbers>
    </Message>
    <Message>
      <MessageText>Sample Message 2</MessageText>
      <Numbers>
        <Number>27825550101</Number>
        <Number>27835550505</Number>
        <Number>27845550909</Number>
      </Numbers>
    </Message>
  </Messages>
</WinSMS_Multi_Message>
```

Messages node: Contains one or more **Message** nodes.

Message node: Repeat once for each unique Message to be sent. There is no limit to the amount of Message nodes.

MessageText node: A message of up to 918 characters. The message will fail if there are more than 918 characters.

NB: To include a carriage return in the Message node please use the URL-encoded value for a line feed: %0A

NB: The following single characters count as 2 characters:

| (pipe)
^
{
}
[
]
~
\

Scheduled node: The Date / Time (**GMT+2**) when the messages must be sent to the mobile device. The format is: yyyyymmddhhmm. If this is equal to or prior to the current Date / Time, then Message will be sent immediately. Calls with an invalid **Scheduled** parameter will be sent immediately.

NB. This node is not mandatory. If it is left out, or left blank, the message will be sent immediately.

Numbers node: Contains one or more **Number** nodes.

Number node: Repeat once for each recipient. Should contain International Format Numbers. There is no limit to the amount of nodes.

OUTPUTS

No Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:elementname="XMLGateway_Response" type="XMLGateway_ResponseType" />
<xs:complexType name="XMLGateway_Response">
  <xs:annotation>
    <xs:appinfo:docElement="XMLGateway_Response" />
  </xs:annotation>
  <xs:sequence>
    <xs:elementname="SendResult" type="SendResultType"
maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="SendResultType">
  <xs:sequence>
    <xs:elementname="MessageNumber" type="xs:integer" />
    <xs:elementname="NumberID" type="NumberIDType" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="NumberIDType">
  <xs:annotation>
    <xs:appinfo:db:repeated="True" />
  </xs:annotation>
  <xs:sequence>
    <xs:elementname="Number" type="xs:integer" />
    <xs:elementname="ID" type="xs:integer" />
  </xs:sequence>
</xs:complexType>
</xs:schema>
```

Example XML:

```
<XMLGateway_Response>
<SendResult>
  <MessageNumber>1</MessageNumber>
    <NumberID>
      <Number>27825550101</Number>
      <ID>374</ID>
    </NumberID>
    <NumberID>
      <Number>27835550505</Number>
      <ID>375</ID>
    </NumberID>
    <NumberID>
      <Number>27845550909</Number>
      <ID>376</ID>
    </NumberID>
  </SendResult>
</SendResult>
<MessageNumber>2</MessageNumber>
  <NumberID>
    <Number>27825550101</Number>
    <ID>377</ID>
  </NumberID>
  <NumberID>
    <Number>27835550505</Number>
    <ID>378</ID>
  </NumberID>
  <NumberID>
    <Number>27845550909</Number>
    <ID>379</ID>
  </NumberID>
</SendResult>
</XMLGateway_Response>
```

SendResult node: Will occur once for each unique message sent.

MessageNumber node: The position of the unique message in the xml document sent to the gateway.

NumberID node: Will occur once for each number sent.

Number node: Contains the mobile number sent to the gateway.

ID node: Possible Values

[INTEGER VALUE]: The unique id assigned by the gateway to that particular message. The Unique ID can be used with the Status XML to determine the status of the message.

Invalid Username or Password: The Username or Password supplied is not valid.

FAIL:TRIALNOT2SELF: Trial Account Restriction – Message not to own number. [More Info](#)

FAIL:TOOLONG: The message exceeded 918 characters.

FAIL:OPTEDOUT: The recipient has opted out from receiving any further messages from the user.

NOCREDITS: The user submitted too many messages to the queue and ran out of credits.

ACCOUNTLOCKED: The Specified Account is locked.

FAIL:BADDEST: An invalid Number was supplied, or the recipient has been Blacklisted from receiving messages.

FAIL:NO MESSAGE: The **MessageText** node was not present or contained a blank string.

FAIL:NO NUMBER: The **Numbers** node was not present, or the **Numbers** node was present but no **Number** nodes were supplied, or the **Number** node contained a blank string.

OUTPUTS

Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:elementname="XMLGateway_Response"
type="XMLGateway_ResponseType" />
<xs:complexType name="XMLGateway_ResponseType">
<xs:annotation>
<xs:appinfo:docElement="XMLGateway_Response" />
</xs:annotation>
<xs:sequence>
<xs:elementname="Error" type="xs:string"
maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Example:

```
<XMLGateway_Response>
  <Error>Not Valid XML Document</Error>
  <Error>Invalid Date</Error>
</XMLGateway_Response>
```

Error node: May occur many times - once for each violation.

Possible Values:

Not Valid XML Document: The posted Document was not in a valid XML format.

Invalid Date: The Date node was present and contained a non-blank value, but was not in the format 'yyyymmddhhmm'.

Batch Status

Description:

This command submits one or multiple message IDs to get their status. Messages statuses can be retrieved for messages sent up to 90 days previously.

Current Location: Call: **<http://www.winsms.co.za/api/XMLGetStatus.asp>**

HTTP Post an XML file with the following layout (**Use exact capitalisation**):

Get Status requests must be limited to a maximum batch size of 1000 message ID's per call. Status requests for message IDs exceeding this limit will be returned with status code 108 (SKIPPED) and will need to be re-queried in subsequent calls.

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:elementname="WinSMS_StatusRequest" type="WinSMS_StatusRequestType" />
  <xs:complexType name="WinSMS_StatusRequestType">
    <xs:annotation>
      <xs:appinfo:docElement="WinSMS_StatusRequest" />
    </xs:annotation>
    <xs:sequence>
      <xs:elementname="User" type="xs:string" />
      <xs:elementname="Password" type="xs:string" />
      <xs:elementname="MessageIDs" type="MessageIDsType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="MessageIDsType">
    <xs:sequence>
      <xs:elementname="MessageID" type="MessageIDType" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="MessageIDType">
    <xs:sequence>
      <xs:elementname="ID" type="xs:integer" />
      <xs:elementname="CLI" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Example XML:

```
<WinSMS_StatusRequest>
  <User>ianw</User>
  <Password>thegreat</Password>
  <MessageIDs>
    <MessageID>
      <ID>192</ID>
      <CLI>27826520938</CLI>
    </MessageID>
    <MessageID>
      <ID>193</ID>
      <CLI>27826520938</CLI>
    </MessageID>
  </MessageIDs>
</WinSMS_StatusRequest>
```

ID node: The unique id assigned by the gateway to the message you wish to retrieve the status of. This is the ID that is returned by XMLSendSMSMulti.asp.

CLI node: This is the CLI to which the message was originally sent. This is purely for feedback purposes, ie. Whatever you place in this field will be sent back in the result XML. ***An incorrect CLI in this field will NOT influence the resulting XML.***

OUTPUTS

No Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="XMLGatewayStatus_Response"
    type="XMLGatewayStatus_ResponseType" />
  <xs:complexType name="XMLGatewayStatus_ResponseType">
    <xs:annotation>
      <xs:appinfo docElement="XMLGatewayStatus_Response" />
    </xs:annotation>
    <xs:sequence>
      <xs:element name="StatusID" type="StatusIDType"
        maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="StatusIDType">
    <xs:sequence>
      <xs:element name="MessageID" type="xs:integer" />
      <xs:element name="CLI" type="xs:string" />
      <xs:element name="Status" type="xs:string" />
      <xs:element name="StatusDateTime" type="xs:string" />
      <xs:element name="StatusCode" type="xs:integer" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Example XML:

```
<XMLGatewayStatus_Response>
  <StatusID>
    <MessageID>192</MessageID>
    <CLI>27826520938</CLI>
    <Status>NOTFOUND</Status>
    <StatusDateTime>2005/03/18 11:17:11</StatusDateTime>
    <StatusCode>-2</StatusCode>
  </StatusID>
  <StatusID>
    <MessageID>12786</MessageID>
    <CLI>27826520938</CLI>
    <Status>DELIVERED</Status>
    <StatusDateTime>2005/03/18 11:15:49</StatusDateTime>
    <StatusCode>0</StatusCode>
  </StatusID>
</XMLGatewayStatus_Response>
```

StatusID node: Will occur once for each MessageID status requested.

MessageID node: Contains the MessageID sent to the gateway.

CLI node: Contains the CLI sent to the gateway. This is purely for feedback purposes, and whatever value was sent, will appear here.

Status Node: A text description of the status of the message.

Possible Values:

SENT = The message has been sent to the recipient, but no delivery report has been received. Further explanation can be seen in the **STATUSCODE** Node.

NOTFOUND = An incorrect MessageID was sent, the message is still in the queue waiting to be sent, or the message was sent more than 90 days previously.

FAIL = An incorrect username or password was sent in the XML requesting the status.

SENDINGFAILED = The message was not delivered to the recipient. The reason can be ascertained using **STATUSCODE**

DELIVERED = The message was successfully delivered to the recipient.

StatusDateTime Node: The format will always be: **yyyy/mm/dd HH:mm:ss. (GMT+2)**. For **DELIVERED** or **SENDINGFAILED** status, the date and time is when the message reached the recipients phone, or finally failed. For **FAIL** or **NOTFOUND**, the current date and time is returned.

StatusCode Node: An integer value representing the sending status, used predominantly to determine a SENDINGFAILED reason

Possible Values:

-3	Invalid username or password in requesting document
-2	Message not found
-1	Message in queue
0	DELIVERED
100	Sent; no delivery report received
101	Undelivered: expired in transit
102	Undelivered: message was cancelled before being delivered
103	Message could not be delivered
105	Not yet delivered, but accepted for delivery
106	Sent; delivery status not known
107	Undelivered: message was rejected
108	SKIPPED
111	Message could not be sent
201	WinSMS account has insufficient credits
202	WinSMS account is locked
211	Message is too long to send by SMS
212	Recipient has opted out of receiving messages
213	Sending or delivery failed for an unknown reason
214	Number is blacklisted

OUTPUTS

Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="XMLGateway_Response"
    type="XMLGateway_ResponseType" />
  <xs:complexType name="XMLGateway_ResponseType">
    <xs:annotation>
      <xs:appinfo db:docElement="XMLGateway_Response" />
    </xs:annotation>
    <xs:sequence>
      <xs:element name="Error" type="xs:string"
        maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Example:

```
<XMLGateway_Response>
  <Error>Not Valid XML Document</Error>
</XMLGateway_Response>
```

Error node: Will occur only once.

Possible Value:

Not Valid XML Document:

Possible reasons:

- The posted Document was not in a valid XML format.
- One of the ID nodes contained a non numeric character.
- One of the ID nodes was blank.

Delete Scheduled

Description:

This command deletes one or multiple scheduled message/s by submitting their message IDs. This will refund the messages' associated Credits to the account.

Current Location: Call: **<http://www.winsms.co.za/api/XMLDeleteScheduled.asp>**

HTTP Post an XML file with the following layout (**Use exact capitalisation**):

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:elementname="WinSMS_DeleteScheduled"
    type="WinSMS_DeleteScheduledType" />
  <xs:complexType name="WinSMS_DeleteScheduledType">
    <xs:annotation>
      <xs:appinfo:docElement="WinSMS_DeleteScheduled" />
    </xs:annotation>
    <xs:sequence>
      <xs:elementname="User" type="xs:string" />
      <xs:elementname="Password" type="xs:string" />
      <xs:elementname="MessageIDs" type="MessageIDsType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="MessageIDsType">
    <xs:sequence>
      <xs:elementname="MessageID" type="MessageIDType"
        maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="MessageIDType">
    <xs:sequence>
      <xs:elementname="ID" type="xs:integer" />
      <xs:elementname="CLI" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Example XML:

```
<WinSMS_DeleteScheduled>
  <User>ianw</User>
  <Password>thegreat</Password>
  <MessageIDs>
    <MessageID>
      <ID>192</ID>
      <CLI>27826520938</CLI>
    </MessageID>
    <MessageID>
      <ID>193</ID>
      <CLI>27826520938</CLI>
    </MessageID>
  </MessageIDs>
</WinSMS_DeleteScheduled>
```

ID node: The unique id assigned by the gateway to the message you wish to delete. This is the ID that is returned by XMLSendSMSMulti.asp.

CLI node: This is the CLI to which the message was originally sent. This is purely for feedback purposes, ie. Whatever you place in this field will be sent back in the result XML. **An incorrect CLI in this field will NOT influence the resulting XML.**

OUTPUTS

No Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:elementname="XMLGatewayDeleteScheduled_Response"
    type="XMLGatewayDeleteScheduled_ResponseType" />
  <xs:complexType name="XMLGatewayDeleteScheduled_ResponseType">
    <xs:annotation>
      <xs:appinfo:docElement="XMLGatewayDeleteScheduled_Response" />
    </xs:annotation>
    <xs:sequence>
      <xs:elementname="DeleteStatus" type="DeleteStatusType"
        maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DeleteStatusType">
    <xs:sequence>
      <xs:elementname="MessageID" type="xs:integer" />
      <xs:elementname="CLI" type="xs:string" />
      <xs:elementname="Status" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Example XML:

```
<XMLGatewayDeleteScheduled_Response>
  <DeleteStatus>
    <MessageID>192</MessageID>
    <CLI>27826520938</CLI>
    <Status>DELETED</Status>
  </DeleteStatus>
  <DeleteStatus>
    <MessageID>193</MessageID>
    <CLI>27826520938</CLI>
    <Status>NOTFOUND</Status>
  </DeleteStatus>
</XMLGatewayDeleteScheduled_Response>
```


DeleteStatus node: Will occur once for each MessageID submitted.

MessageID node: Contains the MessageID sent to the gateway for deletion.

CLI node: Contains the CLI sent to the gateway. This is purely for feedback purposes, and whatever value was sent, will appear here.

Status node: A text description of the delete request status of the message.

Possible Values:

DELETED = The message was successfully deleted, and the credits refunded to the account.

NOTFOUND = The message ID submitted was not found by the WinSMS gateway. This could be because it is incorrect, or because the scheduled message has been deleted previously.

FAIL = An incorrect username or password was sent in the XML requesting the deletion.

SENT = The message ID submitted has already been sent by the WinSMS gateway, and can therefore not be deleted.

OUTPUTS

Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:elementname="XMLGatewayDeleteScheduled_Response"
type="XMLGatewayDeleteScheduled_ResponseType" />
<xs:complexType name="XMLGatewayDeleteScheduled_ResponseType">
<xs:annotation>
<xs:appinfo:docElement="XMLGatewayDeleteScheduled_Response" />
</xs:annotation>
<xs:sequence>
<xs:elementname="Error" type="xs:string"
maxOccurs="1" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Example:

```
<XMLGatewayDeleteScheduled_Response>
  <Error>Not Valid XML Document</Error>
</XMLGatewayDeleteScheduled_Response>
```

Error node: Will occur only once.

Possible Value:

Invalid Username or Password: The Username or Password supplied is not valid.

Not Valid XML Document:

Possible reasons:

- The posted Document was not in a valid XML format.
- One of the ID nodes contained a non numeric character.
- One of the ID nodes was blank.

Replies

Description:

This command returns all replies for a given account. Replies can be retrieved for up to 90 days previously.

Current Location: Call: **<http://www.winsms.co.za/api/XMLGetReplies.asp>**

HTTP Post an XML file with the following layout (**Use exact capitalisation**):

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:elementname="WinSMS_Replies"
    type="WinSMS_RepliesType" />
<xs:complexType name="WinSMS_RepliesType">
<xs:annotation>
<xs:appinfo:docElement="WinSMS_Replies" />
</xs:annotation>
<xs:sequence>
<xs:elementname="User" type="xs:string"
MaxOccurs="1" />
<xs:elementname="Password" type="xs:string"
MaxOccurs="1" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Example XML:

```
<WinSMS_Replies>
  <User>ianw</User>
  <Password>thegreat</Password>
</WinSMS_Replies>
```

User node: WinSMS Login Username.

Password node: WinSMS Login Password.

OUTPUTS

No Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="XMLGateway_Replies"
type="XMLGateway_RepliesType" />
<xs:complexType name="XMLGateway_RepliesType">
<xs:annotation>
<xs:appinfo docElement="XMLGateway_Replies" />
</xs:annotation>
<xs:sequence>
<xs:element name="Reply" type="ReplyType"
maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="ReplyType">
<xs:sequence>
<xs:element name="CLI" type="xs:string" />
<xs:element name="DateReceived" type="xs:string" />
<xs:element name="Message" type="xs:string" />
<xs:element name="SentMessageID" type="xs:integer" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Example XML:

```
<XMLGateway_Replies>
<Reply>
<CLI>27826525555</CLI>
<DateReceived>200401211228</DateReceived>
<Message>This is a reply to your message</Message>
<SentMessageID>235</SentMessageID>
</Reply>
<Reply>
<CLI>27826527676</CLI>
<DateReceived>200401211354</DateReceived>
<Message>I'm replying to your message</Message>
<SentMessageID>241</SentMessageID>
</Reply>
</XMLGateway_Replies>
```

Reply node: Occurs once for each reply received by WinSMS.

CLI node: The mobile number that sent the reply.

DateReceived node: The date that the reply was received by WinSMS. Using the format 'yyyymmddhhmm' (**GMT+2**)

Message node: The reply message received by WinSMS.

SentMessageID node: The unique id assigned by the gateway to the most recent message sent to the mobile number that responded.

NB. Once a reply has been sent via XML, it will not be sent again. It can still, however, be viewed in the WinSMS Client Zone.

OUTPUTS

Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:elementname="XMLGateway_Replies"
type="XMLGateway_RepliesType" />
<xs:complexType name="XMLGateway_RepliesType">
<xs:annotation>
<xs:appinfo:docElement="XMLGateway_Replies" />
</xs:annotation>
<xs:sequence>
<xs:elementname="Error" type="xs:string"
maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Example:

```
<XMLGateway_Replies>
  <Error>Not Valid XML Document</Error>
<Error>Invalid Username or Password</Error>
</XMLGateway_Replies>
```

Error node: Occurs once.

Possible Values:

Not Valid XML Document: The posted Document was not in a valid XML format.

Invalid Username or Password: The Username or Password supplied is not valid.

XML Replies changes:

Introduced 08/06/2016 to accommodate the release of the Short Codes API.

Incoming Short Code Messages should now be retrieved with the Short Code APIs. Please see [page 32-37](#).

To stop receiving short code messages when calling:

<http://www.winsms.co.za/api/XMLGetReplies.asp>

Add the following node to the <WinSMS_Replies> node:

```
<excludeSC>true</excludeSC>
```

THE TAGS ARE CASE SENSITIVE, the 'true' isn't.

So, the full XML might look like this:

```
<WinSMS_Replies>
  <User>ianw</User>
  <Password>thegreat</Password>
  <excludeSC>true</excludeSC>
</WinSMS_Replies>
```

WinSMS HTTP Gateway API

Following is a list of available HTTP commands with their OUTPUTS and examples thereof.

Note: The http request needs to be URL encoded – for more information on URL encoding see http://www.w3schools.com/TAGS/ref_urlencode.asp

Authentication

Description:

This command checks to see if the user name and password to be used is correct and that the account is not locked.

Command:

<https://www.winsms.co.za/api/AUTH.ASP?User=john&Password=John>

Outputs:

Login=OK	Valid Username & Password
Login=Fail	Invalid Username & Password
Login=ACCOUNTLOCKED	Account Locked

Credit Balance

Description:

This command checks the SMS credit balance for the user account.

Command:

<https://www.winsms.co.za/api/credits.ASP?User=john&Password=John>

Outputs:

Credits=Value

Credits=Fail

Invalid Username & Password

[Send Batch](#)

Description:

This command submits one message to a batch of cell numbers. It is also used for sending individual messages.

Command:

<https://www.winsms.co.za/api/batchmessage.asp?User=john&Password=john&Message=Test&Numbers=2783333333;2782222222&Scheduled=200210291230>

Notes:

The Message parameter can be a maximum of 918 characters long. If the Message parameter is longer than 918 characters, the message will fail.

Scheduled is the Date & Time the messages must be sent by the gateway. **(GMT+2)**

Scheduled format is: `yyyymmddhhmm` **(GMT+2)**

Do not use Scheduled if the messages are to be sent immediately.

Calls with an invalid Scheduled parameter will be sent immediately.

NB: To include a carriage return in the Message parameter please use the URL-encoded value for a line feed: %0A

The length of any single HTTP request to the HTTP API is restricted. Please restrict individual calls to the pages to 100 recipients or less.

OUTPUTS:

If the send is successful the following will be returned: `2783333333=52&2782222222=54&`

This output indicates both messages were successfully sent since each message (cell number) is returned with a message ID.

FAIL Invalid Username & Password

Example: `FAIL&`

ACCOUNTLOCKED Account Locked

Example: `2783333333=ACCOUNTLOCKED&2782222222=ACCOUNTLOCKED&`

TRIALNOT2SELF Invalid Cell Number or Cell Number has been Blacklisted

Example: `2783333333=52&2792222222=TRIALNOT2SELF&`

Here the first message was sent (message ID of 52 returned). The second message was not sent because a Trial Account is being used and the recipient number is not the cell number registered to the account.

[More Info.](#)

INSUFFICIENT CREDITS Insufficient Credits

Example: 2783333333=52&2782222222=INSUFFICIENT CREDITS&

Here the first message was sent (message ID of 52 returned), the second message was not sent because the account ran out of credits.

TOOLONG The Message parameter contained more than 918 characters.

Example: 2783333333=TOOLONG&2792222222=TOOLONG&

Here both messages failed because the message was too long.

BADDEST Invalid Cell Number or Cell Number has been Blacklisted

Example: 2783333333=52&2792222222=BADDEST&

Here the first message was sent (message ID of 52 returned), the second message was not sent because it is an invalid cell number.

OPTEDOUT The recipient has opted out from receiving SMSs from your account.

Example: 2783333333=52&2782222222=OPTEDOUT&

Here the first message was sent (message ID of 52 returned), the second message was not sent because the recipient has opted out from receiving messages from your account.

Batch Status

Description:

This command submits one or multiple message IDs to get their status. Messages statuses can be retrieved for messages sent up to 90 days previously.

Command:

<https://www.winsms.co.za/api/RequestBatch.ASP?User=john&Password=John&MessageID=52;54;>

NB. The **MessageID** parameter must end with a ";"

Outputs:

If the MessageID is found, the result is separated into 4 facets by semicolons:

1. The Message **status**.
2. The **status date** (yyyy/mm/dd).
3. The **status time** (hh:mm:ss). (**GMT+2**)
4. The **status code**.

Each **status** is separated by "&", each facet of the **status** is separated by ";"

Possible values for 'status' facet:

SENT	The message has been sent to the recipient, but no delivery report has been received. Further explanation can be seen in STATUSCODE.
FAIL	Invalid Username & Password
Example:	FAIL&
NOTFOUND	Invalid Message ID, Message still in queue, OR Message sent more than 90 days previously.
Example:	52=SENT;2002/02/27;09:29:16;0&54=NOTFOUND& This output shows that message ID 52 was sent at 09:29:16 on 27/02/2002 and that message ID 54 was not found.
DELIVERED	Message was successfully delivered to the recipient.
Example:	54=DELIVERED;2002/02/27;9:29:16;0& This output shows that message ID 54 was delivered at 09:29:16 on 27/02/2002.

Possible values for 'status code' facet:

0	DELIVERED
100	Sent; no delivery report received
101	Undelivered: expired in transit
102	Undelivered: SMS was cancelled before being delivered
103	Message could not be delivered
105	Not yet delivered, but accepted for delivery
106	Sent; delivery status not known
107	Undelivered: message was rejected
108	SKIPPED
111	Message could not be sent
201	WinSMS account has insufficient credits
202	WinSMS account is locked
211	Message is too long to send by SMS
212	Recipient has opted out of receiving messages
213	Sending or delivery failed for an unknown reason
214	Number is blacklisted

Delete Scheduled

Description:

This command deletes one or multiple scheduled message/s by submitting their message IDs. This will refund the messages' associated Credits to the account.

Command:

<https://www.winsms.co.za/api/DeleteScheduled.asp?User=john&Password=john&MessageID=52;54>

Outputs:

DELETED

The message was successfully deleted, and the credits refunded to the account.

Example:

52=DELETED&54=DELETED&

Here both messages were successfully deleted, and the credits refunded to the account.

FAIL

Invalid Username and/or Password. Only one instance of 'FAIL' will be returned, regardless of how many message IDs were submitted.

Example:

FAIL&

NOTFOUND

The message ID submitted was not found by the WinSMS gateway. This could be because it is incorrect, or because the scheduled message has been deleted previously.

Example:

52=DELETED&54=NOTFOUND&

Here the first message was successfully deleted, and the credits refunded. The second message was not found by the WinSMS gateway.

If you had to resubmit the command, you would receive the following response:

52=NOTFOUND&54=NOTFOUND&

As the message with ID = 52 was deleted by the previous command.

SENT

The message ID submitted has already been sent by the WinSMS gateway, and can therefore not be deleted.

Example:

52=SENT&54=SENT&

Here both messages have already been sent by the WinSMS gateway.

Replies

Description:

This command returns all replies for a given account. Replies can be retrieved for up to 90 days previously.

Command:

<https://www.winsms.co.za/api/HTTPGetReplies.ASP?User=john&Password=John>

Outputs:

FAIL Invalid Username & Password

Example: FAIL&

If there are replies the following will be returned.

CLI	The cell number the reply was received from.
DateReceived	The date and time the reply was received (YYYYMMDDHHMM) (GMT+2)
Message	The reply message.
SentMessageID	The unique id assigned by the gateway to the most recent message sent to the mobile number that responded.

Example:

CLI=27831111111;DateReceived=200410071809;Message=test
reply;SentMessageID=5132&CLI=27832222222;DateReceived=200411121117;Message=kewl well
done;SentMessageID=5218&CLI=27833333333;DateReceived=200411302110;Message=Thanx 4
your sms. Had a great day at gold reef city;SentMessageID=12750&

Notes:

Each reply is separated by "&", each facet of the reply is separated by ";".

NB. Once a reply has been sent via HTTPGetReplies.ASP, it will not be sent again. It can still, however, be viewed in the WinSMS Client Zone.

HTTP Replies changes:

Introduced 08/06/2016 to accommodate the release of the Short Codes API.

Incoming Short Code Messages should now be retrieved with the Short Code APIs. Please see [page 32-37](#).

To stop receiving short code messages when calling:

<https://www.winsms.co.za/api/HTTPGetReplies.ASP?User=john&Password=john>

Add the following parameter: &excludesc=true

This is NOT case sensitive.

So, a full URL might look like this:

<https://www.winsms.co.za/api/HTTPGetReplies.ASP?User=john&Password=john&excludesc=true>

List Opt-Outs

Definition:

WinSMS opt-outs are WASPA compliant (<http://www.waspa.org.za>).

An opt-out is registered when an account messages a recipient through the WinSMS gateway, using any WinSMS product or service, and the recipient replies with one of the following (**not** case sensitive):

Message is "Stop"

Message is "End"

Message is "Cancel"

Message is "Unsubscribe"

Message is "Quit"

Message starts with "Stop send"

Message starts with "Opt out"

Message starts with "Opt*out" where '*' represents any character

WinSMS automatically sends a message from the original WinSMS account, confirming the opt-out request, and blocks that WinSMS account from sending any further messages to the particular number.

Opt-out numbers can also be added manually from the WinSMS Client Zone. This functionality can be used to prevent an account from sending a message to a particular number that has **not** specifically opted-out.

Manually added opt-out numbers do not send an opt-out confirmation SMS message to the opted-out number.

Description:

This command returns all opt-out messages received for a given account, as well as all manually added opt-out numbers. There is no start date limit on opt-out messages returned. **All** opt-out messages will be returned with every call.

Command:

<https://www.winsms.co.za/api/ListOptouts.asp?User=john&Password=John>

Outputs:

FAIL Invalid Username & Password

Example: FAIL&

If there are opt-outs the following will be returned.

CLI The cell number that was opted-out (automatically or manually).

DateReceived The date and time the reply was received, or the number added manually in the Client Zone (YYYYMMDDHHMM) **(GMT+2)**

Message The reply message that triggered the automatic opt-out. This field will be * MANUALLY ADDED BY USER * for opt-out numbers added in the Client Zone.

SentMessageID The unique id assigned by the gateway to the most recent message sent to the mobile number that opted-out. This will be an empty string for opt-out numbers added in the Client Zone.

Example:

CLI=2783111111;DateReceived=201706121343;Message=* MANUALLY ADDED BY USER
*;SentMessageID=&CLI=2783222222;DateReceived=201706121329;Message=Optout;SentMessageID=6731673&

Notes:

Each opt-out record is separated by "&", each facet of the record is separated by ";".

NB. Records returned by this API call will also be returned by the "Replies" API call, if the opt-out numbers were not manually added in the Client Zone.

Credit Transfer

Description:

This command transfers credits between accounts using the main accounts credentials. The following credit transfers are possible:

- From main account to sub account.
- From sub account to main account.
- From sub account to another sub account.

Your AccountId is your WinSMS account number. Your WinSMS account number and sub account number/s can be obtained by logging to the WinSMS client zone (www.winsms.co.za/cz) with the main accounts credentials. The main account number is on the home tab and the sub account numbers are under the sub accounts tab.

Command:

<https://www.winsms.co.za/api/creditsTransfer.asp?User=john&Password=john&fromAccountId=xxx&toAccountId=xxxx&transferQuantity=1>

Outputs:

SUCCESS

The credit transfer was successful.

Example:

Transfer=Success

The credits have been successfully transferred from the specified **from account** to the specified **to account**.

FAIL

The credit transfer failed, please check the response.

Example:

Transfer=Fail

The transfer failed due to an unknown reason

Example:

Transfer=invalidUser

Invalid Username and/or Password

Example:

Transfer=invalidFromAccount

The **from account** is not associated with the main account credentials.

Example:

Transfer=invalidToAccount

The **to account** is not associated with the main account credentials.

Example:

Transfer=lockedFromAccount

The **from account** is locked and credit transfers are not allowed.

Example:

Transfer=lockedToAccount

The **to account** is locked and credit transfers are not allowed.

Example:

Transfer=sameAccount

You cannot transfer in and out of the same account.

Example:

Transfer=invalidQuantity

The transfer failed due to there being an invalid number in the **transferQuantity** value. This must be a valid positive integer value. No decimal points allowed.

Example:

Transfer=insufficientCredits

The **from Account** does not have sufficient credits to do the transfer.

WinSMS FTP Gateway API

Current Location: **ftp://ftp.winsms.co.za**

FTP an ASCII file:

Two file formats are supported:

1. Comma Delimited With Quotes (Required Fields in Bold):

Line 1: "**UserName**", "**Password**", "200602241900"
Line 2: "**CellNumber**", "**Message**", "RecipientFirstName", "RecipientSurname"
Line 3: "**CellNumber**", "**Message**", "RecipientFirstName", "RecipientSurname"
Line 4: "**CellNumber**", "**Message**", "RecipientFirstName", "RecipientSurname"
Etc ...

2. Comma Delimited Without Quotes (Required Fields in Bold):

Line 1: **UserName**, **Password**, 200602241900
Line 2: **CellNumber**, **Message**, RecipientFirstName, RecipientSurname
Line 3: **CellNumber**, **Message**, RecipientFirstName, RecipientSurname
Line 4: **CellNumber**, **Message**, RecipientFirstName, RecipientSurname
Etc ...

Notes:

- a) Do not include line numbers.
- b) An FTP access account must first be requested and set up on the gateway in order to gain access to FTP root directory.
- c) If using the file format Comma Delimited WITHOUT Quotes, be sure not to include any commas in the body of the Message field.
- d) To schedule the batch for later delivery, include the date/time (**GMT+2**) on the first line.

Example:

"UserName", "Password", "201202241900"

OR

UserName, Password, 201202241900

This batch will be scheduled for 19H00 on 24 February 2012.

NB. Any individual messages longer than 918 characters will be truncated to 918 characters. You will not be notified of this truncation. This is because large FTP files could contain thousands of messages that are too long, and a notification for each message would be impractical. Please ensure that each individual message in the file is no longer than 918 characters (taking into account the characters that count as 2 characters, mentioned on page 1)

WinSMS E-mail to SMS Gateway

NB! The E-mail to SMS gateway is not to be used for Bulk SMS (batches > 100). Bulk SMS should be sent using one of the above API methods, the desktop client software, or WinSMS.mobi.

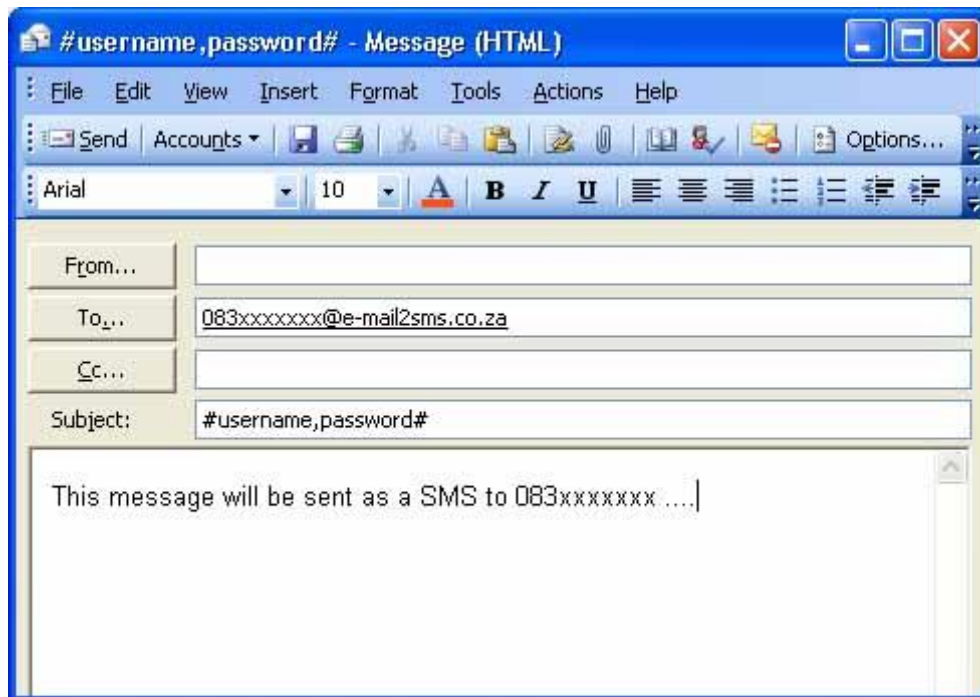
The WinSMS E-mail to SMS Gateway allows one to send e-mails which are then converted to SMS messages.

The WinSMS E-mail to SMS Gateway supports two different methods of authentication:

- 1. Subject Authentication** - You include your WinSMS username and password in the Subject of the e-mail.
 - 2. Sender Address Authentication** - You specify the e-mail addresses from which you want to be able to send e-mail to SMS messages.
-

Sending E-mail to SMS using Subject Authentication

Simply send an e-mail to **cellNumber@e-mail2sms.co.za** with your WinSMS username and password in the subject field using the format: **#username,password#**



The username and password must be separated by a comma, and must be enclosed by a hash (#).

The gateway will send the first **160/918 characters** of the body of the e-mail as the SMS message.

Scheduling:

If you want to schedule the message for later delivery, include the date/time **(GMT+2)** in the subject line in the following format:

Subject: #john,password,YYYYMMDDHHMM#

YYYYMMDDHHMM represents the date and time **(GMT+2)** the message should be sent – for example:

Subject: #john,password,201311011500#

In this example the message will be scheduled for 1st November 2013 at 15:00 (3PM).

To view or download your logs of sent and scheduled messages just login to the Client Zone and go to "Logs".

[Sending E-mail to SMS using Sender Address Authentication](#)

To be able to use Sender Address Authentication you must first specify which e-mail addresses can be used for authentication.

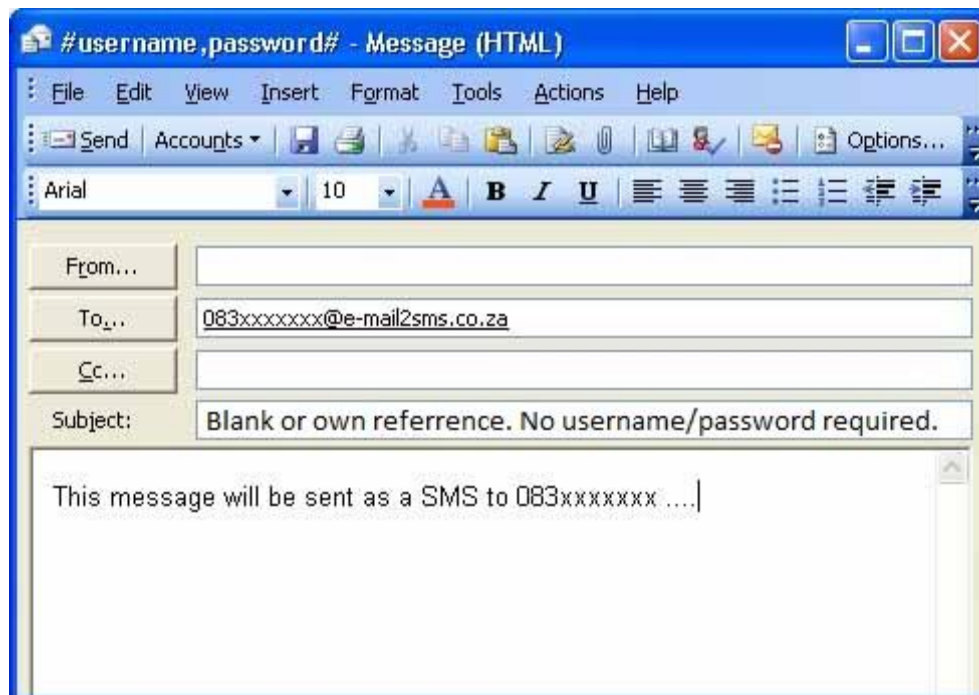
To do this log into the Client Zone and navigate to the "Email to SMS" tab of the "My Profile" page where you can add one or more e-mail addresses as shown below:

Sender E-mail Address Authentication ?

+ Add Email Address	
Email Address ^	
support@winsms.co.za	X
winsms@winsms.co.za	X

Once you have done this you will be able to send e-mail to SMS messages simply by addressing your e-mail to

cellNumber@e-mail2sms.co.za



The screenshot shows an email client window titled "#username,password# - Message (HTML)". The window has a menu bar (File, Edit, View, Insert, Format, Tools, Actions, Help) and a toolbar with icons for Send, Accounts, and various editing functions. Below the toolbar is a rich text editor with a font dropdown set to "Arial" and a size dropdown set to "10". The message fields are as follows:

- From...**: Empty text box.
- To...**: 083xxxxxxx@e-mail2sms.co.za
- Cc...**: Empty text box.
- Subject:** Blank or own reference. No username/password required.

The message body contains the text: "This message will be sent as a SMS to 083xxxxxxx|"

The username and password are NOT required in the subject. The subject can be used for any personal reference and it won't appear in the SMS that gets sent.

The gateway will send the first **160/918 characters** of the body of the e-mail as the SMS message.

Scheduling:

If you want to schedule the message for later delivery, include the date/time **(GMT+2)** in the subject line in the following format:

Subject: YYYYMMDDHHMM

YYYYMMDDHHMM represents the date and time the message should be sent – for example:

Subject: 200904011500

In this example the message will be scheduled for 1st April 2009 at 15:00 (3PM).

To view or download your logs of sent and scheduled messages just login to the client zone and go to "Logs".

Message length (160/918 characters):

You can change the number of e-mail message characters that are sent as an SMS from **160** to **918** characters by enabling long messages.

To do this log into the Client Zone and navigate to the "E-mail to SMS" section at the bottom of the "My Profile" page, then check the **Enable Long Messages** checkbox

NB

This will allow 918 characters of ALL emails that you send to the Gateway to be sent as SMS messages.

Some email software and servers will append signatures, anti-virus information and disclaimers to your mail message that are not visible to you at the time of sending your mail.

If Long Messages are enabled, these characters will be included in the SMS, resulting in up to 6 times more credits being used per message.

Only enable this option if you are fully aware of exactly what text will be sent to the WinSMS Gateway.

TRIAL ACCOUNT RESTRICTIONS

WinSMS Trial Accounts are restricted to sending SMS messages to their own registered cell number. You cannot send SMS messages to cell numbers other than your own.

If you attempt to use the WinSMS E-mail to SMS Gateway to send an SMS to a cell number other than your own, the message will not be delivered, and you will be notified via e-mail of the delivery failure.

[More information on Trial Accounts.](#)

WinSMS Outlook Mobile Service API

NB! Support for Microsoft Office Outlook Mobile Services (OMS) has been removed from Microsoft Outlook 2013.

Whilst WinSMS will continue to provide Outlook Mobile Service (OMS) for previous versions on Microsoft Outlook, if you are using Microsoft Outlook 2013 you will no longer have the option to use Outlook Mobile Services (OMS).

Overview

With the WinSMS Outlook Mobile Service and Microsoft's Outlook 2007 and 2010 versions, it is now as easy to send and receive SMS messages as it is to send and receive email, and you don't need to install any software!

Once you have added the Outlook Mobile Service Account, all you need to do is click on **New** and choose **Text Message**...

Alerts & Reminders

Not only can you send and receive SMS just like email, you can also configure the Outlook Mobile Service to SMS your calendar and appointment reminders, as well as have certain emails automatically forwarded to your cell phone...

Outlook Mobile Notification Options

Message Notification:

- Forward incoming messages as text messages to one or more cell numbers.
- Specify various conditions or rules to use.

Reminders and Calendar Alerts:

- Send Reminders and Calendar Summary to your cell phone.
- Send next day's schedule to your mobile at the end of every day.
- Send appointment or meeting requests to your mobile.

Messages of up to **918 characters** will be sent by the WinSMS Outlook Mobile Service, messages longer than this will be truncated.

TRIAL ACCOUNT RESTRICTIONS

WinSMS Trial Accounts are restricted to sending SMS messages to their own registered cell number.

You will receive a delivery failure if you attempt to use the WinSMS Outlook Mobile Service API to send SMS messages to cell numbers other than your own, when using a Trial Account.

[More information on Trial Accounts.](#)

How to Configure the WinSMS Outlook Mobile Service for Outlook 2007

Once you have registered an account with WinSMS you need to configure Outlook 2007 to use that account.

- **Step 1**
Open Outlook 2007, click on **Tools** and select **Account Settings**.
- **Step 2**
Click on the **New** button.
- **Step 3**
Select **Other** and then select **Outlook Mobile Service** and proceed by clicking Next.
- **Step 4**
Copy and paste the following into the **Web Service Address** box:

<https://www.winsms.co.za/oms/omsservice.asmx>

Under **User ID** and **Password** enter your WinSMS login name and your WinSMS password.

- **Step 5**
Click on **More Settings** to verify that Outlook 2007 was able to connect to the WinSMS Outlook Mobile Service.

Check under **Other User Information** to verify that your **cell number** is correct.

(If your cell number is not correct you will need to log in to the WinSMS Client Zone and go to My Profile where you will be able to specify your correct cell number.)
Click on **OK** to go back to the previous screen.

- **Step 6**
Click on **Test Account Settings** – this will instruct Outlook 2007 to send a test message to your cell phone.

How to Configure the WinSMS Outlook Mobile Service for Outlook 2010

Once you have registered an account with WinSMS you need to configure Outlook 2010 to use that account.

- **Step 1**
Open Outlook 2010, click on **File > Info** and then select **Account Settings**.

- **Step 2**
Click on the **New** button.

- **Step 3**
Select **Text Messaging (SMS)** and then click on **Next**.

- **Step 4**
Copy and paste the following into the **Service Provider URL** box:

<https://www.winsms.co.za/oms/omsservice.asmx>

Under **User ID** and **Password** enter your WinSMS login name and your WinSMS password.

- **Step 5**
Click on **More Settings** to verify that Outlook 2010 was able to connect to the WinSMS Outlook Mobile Service.

Check under **Other User Information** to verify that your **cell number** is correct.

(If your cell number is not correct you will need to log in to the WinSMS Client Zone and go to My Profile where you will be able to specify your correct cell number.)

Click on **OK** to go back to the previous screen.

- **Step 6**
Click on **Test Account Settings** – this will instruct Outlook 2010 to send a test message to your cell phone.

Standard Rate Short/Long Code

Overview

This is fully defined as: Standard rate short/long code for MO (Mobile Originating) SMS.

This service enables you to receive incoming SMS messages sent from a mobile phone to a 5 digit short code or a 10 digit long code assigned to you by WinSMS.

Each short/long code assigned requires a once off setup fee, and is rented to you for a monthly subscription fee.

Incoming SMS messages are billed to the sender at standard network rates (ie. This IS NOT a premium rated short/long code service).

The incoming messages are stored in a log on the WinSMS servers.

This log can be viewed and downloaded by you after logging on to the WinSMS Client Zone.

You can retrieve your incoming short/long code messages via HTTP or XML.

In addition to the Short/Long Code Log in the WinSMS Client Zone, the HTTP and XML Incoming Short/Long Code Message APIs documented below, there are various routing options available. These routing options can be applied by you to enable a more proactive approach to retrieving and responding to incoming SMS messages.

These routing options can be managed by you after logging on to the WinSMS Client Zone.

They are detailed below:

Forward to Email

If this option is enabled, the details of the incoming SMS message will immediately be sent to the email address specified. You can forward incoming SMS messages to a maximum of 20 email addresses, separated by a comma (,).

The following details of the incoming SMS message will be included:

- Date and Time that the message was received. **(GMT+2)**
- The sender's mobile number.
- The message that was sent.

Forward to SMS

If this option is enabled, the details of the incoming SMS message will immediately be sent to the mobile number specified. You can forward incoming SMS messages to a maximum of 20 mobile numbers, separated by a comma (,).

The following details of the incoming SMS message will be included:

- The message that was sent.
- The sender's mobile number.

The cost of the SMS to the number specified will be deducted from your WinSMS credits. If you have 0 credits, the SMS will not be sent.

To ensure that this never happens, it is best to set the "Credit Notification Level" under "My Profile" in the Client Zone.

Auto Reply SMS

If this option is enabled, a static response SMS will immediately be sent to the sender's mobile number.

This response will be specified by you, and can be a maximum of 918 characters long.

The cost of the response SMS will be deducted from your WinSMS credits. If you have 0 credits, the response SMS will not be sent.

To ensure that this never happens, it is best to set the 'credit notification level' under your 'profile' in the Client Zone.

Forward to URL

This advanced option allows developers to further integrate incoming SMS messages, and customise responses.

By creating and hosting a dynamic web page (written in asp.NET, perl, asp, PHP etc.), you can have incoming SMS messages interact with your internal systems, and dynamically create appropriate responses to the sender. Customised responses can be sent from your dynamic web page using the WinSMS HTTP or XML gateway.

To enable 'Forward to URL' functionality, you must first develop the dynamic web page, and then host it on a server. Once this is done, you can test that your page is working correctly using the tools in the Client Zone, and then save the page URL in the 'Forward to URL' field.

Every time a new incoming SMS message is received by WinSMS, the specified URL will immediately be called with the following parameters (using a HTTP/S GET request):

- Date and Time that the message was received. **(GMT+2)**
- The sender's mobile number.
- The message that was sent.

Input / Output details

Parameters sent using HTTP/S GET to the URL specified:

- **date** - Message time stamp, formatted YYYYMMDD-HHMMSS. **(GMT+2)**
- **from** - Telephone number from which the message was sent.
- **text** - Text of the message.

The dynamic web page called should generate an HTML response page confirming success or describing an error, using the following format:

Success:

```
<html>
<body>
returnCode=OK
</body>
</html>
```

Error:

```
<html>
<body>
returnCode=KO
errorCode=500
errorDescription=Internal server error
</body>
</html>
```

The HTML tags are purely for legibility while you are testing. The only requirement of the WinSMS gateway is that the following text appears somewhere in the response (case and space sensitive):

returnCode=

You will not be able to save changes to the 'Forward to URL' field unless a valid response is returned by your page.

Please test your page, to ensure a valid response, before saving changes.

Testing 'Forward to URL' setting:

To test the dynamic web page that you have created, log on to the WinSMS Client Zone, and go to the short/long code configuration page.

Here you can enter the URL of your page in the 'Forward to URL' field. Remember to include 'http://' or 'https://' depending on the protocol you are using.

Click the 'Test' button in the 'Forward to URL' section.

A new window or tab will be opened in your web browser, displaying the results of the test.

The URL specified will be called with 3 dummy parameters.

- Date=20130107-153552
- From=27825555555
- Text=Sample Text from WinSMS Client Zone

At the top of the new window, the following will be displayed:

- The URL specified in the 'Forward to URL' field.
- The result of the call to the URL specified.

The result will be one of the following:

- **Success** – The URL is valid, and returns a valid response.
- **Invalid URL specified for callback** – The URL specified does not exist, or an invalid protocol was specified, or no protocol was specified.
- **Invalid Response from URL callback page** – The URL exists, but an invalid response was returned. Ie. '**returnCode=**' was not in the response.

In the lower section of the new window, the complete response from the URL specified is displayed.

If you are in the process of developing a page, this display could be handy, as you can include debug information in the html response generated.

Requesting Incoming Short/Long Code Messages via HTTP

Short/Long code messages can be retrieved for up to 90 days previously.

Description:

This command returns all short/long code messages for a given account.

Command:

<https://www.winsms.co.za/api/HTTPGetSCmessages.asp?User=john&Password=John>

Outputs:

FAIL Invalid Username & Password

Example: FAIL&

If there are replies the following will be returned.

SC	The short/long code number.
CLI	The cell number the reply was received from.
DateReceived	The date and time the reply was received (YYYYMMDDHHMM) (GMT+2)
Message	The reply message.

Example:

SC=43212;CLI=27606679266;DateReceived=201606081137;Message=Test to
43212&SC=43213;CLI=27606679266;DateReceived=201606081137;Message=Test 2 to
43213&SC=43212;CLI=27606679266;DateReceived=201606081138;Message=Test to
43212&SC=43212;CLI=27606679266;DateReceived=201606081138;Message=Test 2 to 43212&

Notes:

Each short/long code message is separated by "&", each facet of the short/long code message is separated by ";".

NB. Once a short/long code message has been sent via HTTPGetSCmessages.asp, it will not be sent again. It can still, however, be viewed in the WinSMS Client Zone.

Requesting Incoming Short/Long Code Messages via XML

Current Location: Call: **http://www.winsms.co.za/api/XMLGetSCmessages.asp**

Short/Long code messages can be retrieved for up to 90 days previously.

HTTP Post an XML file with the following layout (**Use exact capitalisation**):

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="WinSMS_SCmessages"
    type="WinSMS_SCmessagesType" />
  <xs:complexType name="WinSMS_SCmessagesType">
    <xs:annotation>
      <xs:appinfo db:docElement="WinSMS_SCmessages" />
    </xs:annotation>
    <xs:sequence>
      <xs:element name="User" type="xs:string"
        MaxOccurs="1" />
      <xs:element name="Password" type="xs:string"
        MaxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Example XML:

```
<WinSMS_SCmessages>
  <User>ianw</User>
  <Password>thegreat</Password>
</WinSMS_SCmessages>
```

User node: WinSMS Login Username.

Password node: WinSMS Login Password.

OUTPUTS

No Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="XMLGateway_SCmessages"
    type="XMLGateway_SCmessagesType" />
  <xs:complexType name="XMLGateway_SCmessagesType">
    <xs:annotation>
      <xs:appinfo docElement="XMLGateway_SCmessages" />
    </xs:annotation>
    <xs:sequence>
      <xs:element name="SCmessage" type="SCmessageType"
        maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="SCmessageType">
    <xs:sequence>
      <xs:element name="SC" type="xs:string" />
      <xs:element name="CLI" type="xs:string" />
      <xs:element name="DateReceived" type="xs:string" />
      <xs:element name="Message" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Example XML:

```
<XMLGateway_SCmessages>
  <SCmessage>
    <SC>43212</SC>
    <CLI>27826525555</CLI>
    <DateReceived>200401211228</DateReceived>
    <Message>This is a test message to 43212</Message>
  </SCmessage>
  <SCmessage>
    <SC>43212</SC>
    <CLI>27826527676</CLI>
    <DateReceived>200401211354</DateReceived>
    <Message> This is another test message to 43212</Message>
  </SCmessage>
</XMLGateway_SCmessages>
```

SCmessage node: Occurs once for each short/long code message received by WinSMS.

SC node: The short/long code number.

CLI node: The mobile number that sent the reply.

DateReceived node: The date that the short/long code message was received by WinSMS. Using the format 'yyyymmddhhmm' (**GMT+2**)

Message node: The short/long code message received by WinSMS.

NB. Once a short/long code message has been sent via XML, it will not be sent again. It can still, however, be viewed in the WinSMS Client Zone.

OUTPUTS

Fatal Errors Encountered: The Gateway will return XML with the following layout:

Schema:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:elementname="XMLGateway_SCmessages"
type="XMLGateway_SCmessagesType" />
<xs:complexType name="XMLGateway_SCmessagesType">
<xs:annotation>
<xs:appinfoxdb:docElement="XMLGateway_SCmessages" />
</xs:annotation>
<xs:sequence>
<xs:elementname="Error" type="xs:string"
maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Example:

```
<XMLGateway_SCmessages>
  <Error>Not Valid XML Document</Error>
<Error>Invalid Username or Password</Error>
</XMLGateway_SCmessages>
```

Error node: Occurs once.

Possible Values:

Not Valid XML Document: The posted Document was not in a valid XML format.

Invalid Username or Password: The Username or Password supplied is not valid.

Reply Notification and Delivery Report Callbacks

Overview

This advanced option allows developers to further integrate incoming SMS replies and delivery/failure reports into their existing systems. Eg. The systems that initially sent the outgoing message.

By creating and hosting a dynamic web page (written in asp.NET, perl, PHP, Node.js etc.), you can have reply messages and delivery reports sent asynchronously from WinSMS, interact with your internal systems, and dynamically create appropriate system responses. Eg Send a response to an incoming message, or retry or blacklist a failed delivery.

To enable callback functionality, you must first develop the dynamic web page, and then host it on a server. Once this is done, you can test that your page is working correctly using the tools in the Client Zone, then save the page URL in the 'Callback URL' field on the Developers Tab of the 'My Profile' section in the Client Zone.

Reply Notification Callbacks and Delivery Report Callbacks can be enabled separately on the Developers tab of the Client Zone. Although BOTH callbacks will be delivered to the same URL.

They can be differentiated by the 'type' parameter in the GET post to the URL.

Delivery Reports will be represented by a 'type' parameter of '**report**' and Replies will be represented by a 'type' parameter of '**deliver**'.

Reply Notification Callbacks

If Reply Notification Callbacks are enabled in the Client Zone, the following will happen:

Every time a reply is received by WinSMS, the specified callback URL will immediately be called with the following parameters (using an HTTP/S GET request):

- Date and Time that the message was received. (**GMT+2**)
- The sender's mobile number.
- The message that was sent.

Exact parameters sent by WinSMS, using HTTP/S GET, to the URL specified:

- **type** – This will always be '*deliver*' for Reply Callbacks
- **date** - Message time stamp, formatted YYYYMMDD-HHMMSS. (**GMT+2**)
- **from** - Telephone number from which the message was sent.
- **text** - Text of the message.
- **sentmessageid** – The unique WinSMS message id assigned to the original OUTGOING message with which the reply is associated.
- **userid** – The unique WinSMS account number (User Id) of the account that sent the original OUTGOING message. This is handy if you use the same Callback URL for multiple WinSMS Accounts or Sub Accounts.

The dynamic web page called should respond with an HTTP Code 200, to indicate that it acknowledges receipt of the callback.

Failure to do so will result in WinSMS attempting the callback a subsequent 3 times.

Delivery Report Callbacks

If Delivery Report Callbacks are enabled in the Client Zone, the following will happen:

Every time a delivery/failure report is received by WinSMS, the specified callback URL will immediately be called with the following parameters (using an HTTP/S GET request):

- Date and Time that the message was delivered / failed. **(GMT+2)**
- Date and Time that the message was sent. **(GMT+2)**
- The mobile number to which the message was sent.
- The status of the message – DELIVERED or FAILED.
- The WinSMS message id originally assigned to the sent message.

Exact parameters sent by WinSMS, using HTTP/S GET, to the URL specified:

- **type** – This will always be *report* for Delivery Report Callbacks
- **date** – The date the message was originally **sent**, formatted YYYYMMDD-HHMMSS. **(GMT+2)**
- **sdate** – The Status Date of the message – the date it was delivered or failed, formatted YYYYMMDD-HHMMSS. **(GMT+2)**
- **from** – The telephone number to which the message was sent.
- **state** – The final state of the message – **DELIVERED, UNDELIV** etc. – See below
- **status** – Integer code representing the current state of the message. Refer to the table below for possible values.
- **sentmessageid** – The unique WinSMS message id originally assigned to the sent message.

The dynamic web page called should respond with an HTTP Code 200, to indicate that it acknowledges receipt of the callback.

Failure to do so will result in the WinSMS attempting the callback a subsequent 3 times.

State and Status Codes:

<u>status</u>	<u>state</u>	<u>Description</u>
0	DELIVERED	
101	EXPIRED	Undelivered: expired in transit
102	DELETED	Undelivered: message cancelled before being delivered
103	UNDELIV	Undelivered; message could not be delivered
107	REJECTD	Undelivered: message was rejected