

*SMS Everyone HTTP Interface*  
**API Documentation v1.46**

## Contents

<i>SMS Everyone HTTP Interface API Documentation v1.46</i> .....	1
Introduction .....	4
Sending and Receiving SMS .....	4
Sending SMS (MT) .....	4
Receiving SMS (MO).....	5
HTTP Parameters to send / receive messages.....	6
Basic Parameters for sending SMS.....	6
Parameters for sending SMS and parameters we send you when receiving SMS on your web service .....	6
Parameters for Scheduling a Campaign for a later date/time .....	9
Parameters for modifying a campaign.....	9
Parameters for deleting a campaign.....	9
Parameters for pausing a campaign .....	9
Parameters for checking status of a Campaign .....	10
Send Status of Message .....	10
2-way (Receiving reply messages on demand) .....	11
Retrieving All Unretrieved Replies .....	11
Retrieving All Replies for a given period .....	12
Checking SMS Credit Balance.....	14
Add Credits.....	15
Optouts .....	16
Add a number to our optout database (opt out / unsubscribe).....	16
Delete a number from our optout database (opt back in / re-subscribe).....	16
Retrieve a list of all numbers that are opted out / unsubscribed.....	17
Extras.....	17
Time zones .....	17
Send Window .....	17
Samples .....	17
Sample for Sending a Text Message immediately .....	17
Sample for Scheduling a Text Message to be sent later .....	17
Sample for modifying a scheduled Campaign.....	17
Sample for deleting a scheduled Campaign.....	17
Sample for checking the status of a scheduled Campaign.....	18
Sample for checking for all unretrieved replies .....	18
Sample for checking for all replies in the last 7 days .....	18

Sample for checking for all replies from April 1 <sup>st</sup> 2020 – April 3 <sup>rd</sup> 2020 from 6402111222.....	18
Replies Samples.....	18
Optouts Samples .....	18
Appendix A: Http Response Codes.....	20
Appendix B: URL Encoding Why do we need URL Encoding? .....	22
How are characters URL Encoded? .....	22
What characters need to be encoded and why? .....	22

# Introduction

This document outlines the interfaces by which SMS messages can be delivered and received over standard HTTP (web) protocol. We support both HTTP POST and GET.

Please note: MMS Messages cannot be delivered via this HTTP interface.

## Sending and Receiving SMS

### Sending SMS (MT)

The SMS Everyone web service allows you to send messages from your website to any mobile phone. This is known as a Mobile Terminated (MT) message.

MT messages are sent by sending an HTTP request (POST or GET) directly to the SMS Everyone web service address:

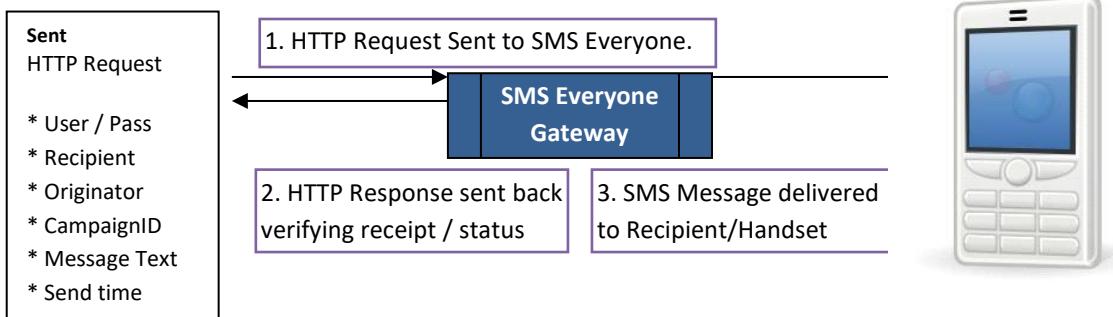
**HTTPS (Recommended)**

<https://api.smseveryone.com/ws/campaign>

**HTTP**

<http://api.smseveryone.com/ws/campaign>

**www.yoursite.com**



### Scheduling Campaigns

This service allows you to schedule SMS's to be sent at a later date/time by passing the date and time you would like the message sent as a parameter in your POST.

You can also delete scheduled campaigns up to 10 seconds prior to the message being sent. See the 'Parameters for Deleting a Campaign' section below.

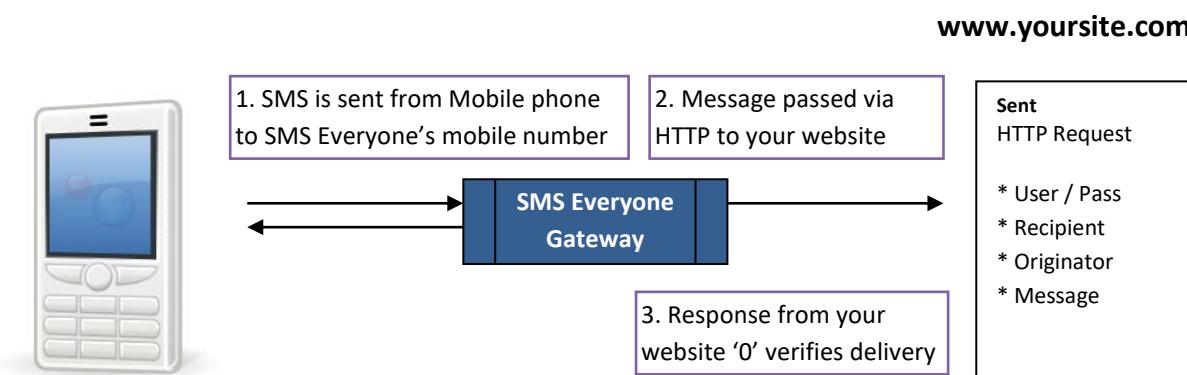
## Receiving SMS (MO)

You can also receive messages from any mobile phone. This is known as a Mobile Originated (MO) message. There are 2 options for receiving replies:

### *OPTION 1 - Receiving real time to your web service*

MO messages are forwarded to your nominated website address (URL) by SMS Everyone via an HTTP GET request. (POST is also available if required) This is mainly used where you have a single URL that we can post all replies to.

A correctly configured page on your web site must be set up to receive messages of this type and you must return the correct HTTP response of 0 (zero).



The fields we pass to you are:

username (if required)

password (if required)

message\_text

originator (the end user's mobile number)

recipient (our virtual mobile number)

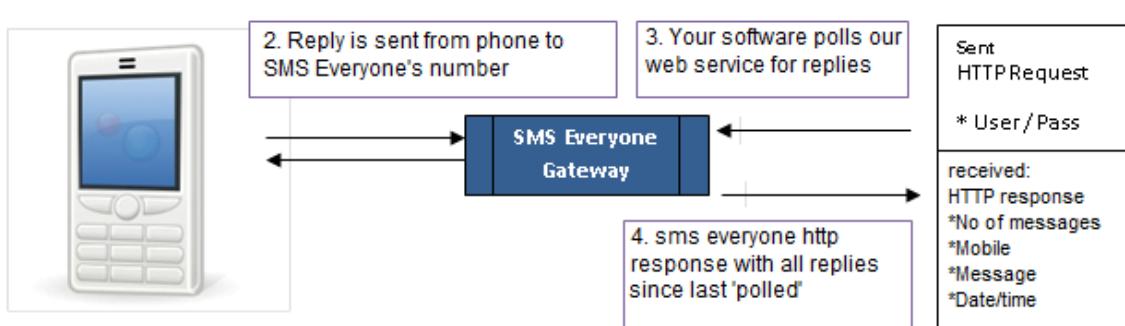
reference (a unique reference ID)

See the samples section later in this document for an example string.

## *OPTION 2 - Receiving messages on demand (2 WAY)*

MO messages are stored on our URL by SMS Everyone. Your system ‘polls’ our web service (via HTTP POST or GET) and retrieves replies in batches as required. We will only pass replies to you that have not yet been retrieved. This system is used where you have multiple instances of your software and you need to get replies to the right account and/or you don’t have a specific URL we can point to. For example: appointment reminder software on multiple client’s computers. This requires the outbound message to have been sent via our MT API so that we can route the reply to the correct account. See ‘2 way’ section below.

1. Outbound message is sent to phone. Eg Appointment reminder



## HTTP Parameters to send / receive messages

The basic HTTP parameters are the same for sending & receiving messages.

The following tables outline the parameters that you will need to include in your http request OR that we will post to you for replies.

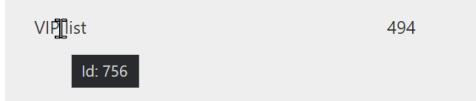
Note that we accept all variations of the parameters listed in the parameters field below. Eg ‘USER\_NAME’, ‘user’, ‘username’, ‘usr’ & ‘name’ are all accepted as the ‘user name’ field name. We provide compatibility against all other major SMS provider API’s. Let us know if you need any other field names than the ones below.

### Basic Parameters for sending SMS

Parameters	Required	Description
username user_name user usr name AccountSid	Yes	Username provided by SMS Everyone for use of the HTTP Interface. The username field is <u>not</u> case sensitive
password pass pwd AuthToken	Yes	Password provided up by SMS Everyone for use of the HTTP Interface. Case sensitive

### Parameters for sending SMS and parameters we send you when receiving SMS on your web service

originator from origin sender	Yes	Phone Number of the Originator or sender of the message.  MT – if the message is from you to a mobile phone, this is the phone number or word origin of your service. Eg ‘JIMS BAR’
--	-----	---

		<p>MO – if the message is from a mobile to your service, this is the mobile phone number that sent the message</p> <p>Max length = 11 alphanumeric characters (ie 0-9, a to z, A to Z)</p> <p>Specifying ~def~ will use your system default Originator.</p> <p>If your originator is a mobile phone number, it is recommended to put it in International format <u>without</u> the + (plus) sign. Eg 61402111222</p>
<b>destination recipient dest to</b>	Yes	<p>Phone number of the recipient.</p> <p>MT - If the SMS is from a Service Provider to a Mobile Phone (Mobile Phone Terminated) then this will contain the number of the Mobile Phone.</p> <p>MO - If the request is from a Mobile Phone to a Service (Mobile Phone Originated) then this will contain the number of the Service provider (also known as a Virtual Mobile number) that the SMS was sent to. (provided by SMS Everyone)</p> <p>Ideally this should also be sent in international format <u>without</u> the + (plus) sign. eg: a standard Australian number = 61402777666 a New Zealand number = 6421575222</p> <p>You can submit either a single number per post for individual messages or for bulk sms you can submit up to 1000 numbers per post, separated by commas. Eg: <code>&amp;destination=61400000001,61400000002,61400000003,etc</code></p> <p>If you include a campaign id in each bulk post, we will combine the numbers together into one campaign, deduplicate them for you at send and wash against any optouts. You can then modify the whole campaign or delete the whole campaign in a single post. (see modifying / deleting a campaign below)</p>
<b>listids</b>	No	<p>You can send to a list that is stored on your SMS Everyone account.</p> <p>Log in to your account using the same username and password that you use for this API. Go to <b>lists</b>, upload your list of mobile numbers, then on the lists page, hover your mouse over the description of the list. You'll see the list ID pop up in a black box.</p>  <p>Submit this list ID in your post and an sms will be sent to everyone in that list. Eg: <code>&amp;listids=1234</code></p>
<b>message messagetext text message_text msg txt body</b>	Yes	<p>The SMS Text message being sent. The total length of the SMS Message should be no more than 160 characters for 1 message.</p> <p>If your Text message is longer than 160 characters, you can just send it in this field as normal and our system will automatically concatenate the message chunks together and pass it to the phone. It will appear on the phone as a single long message.</p> <p>Messages over 160 characters are charged as 2, 3 etc</p> <p>Please note:</p> <p>The maximum length of a double message is 306 characters, not 320 characters (160 x 2) as you might expect.</p> <p>The length of the body of a multi-part SMS is reduced from 160 to 153 because those extra 7 characters are needed to hold 'segmentation information' that is sent along with the body of the sms, so that the handset knows how to put all the parts back together again into a single long message.</p>

		<p>We can enable or disable the ability for you to send longer messages. If disabled, we will simply truncate the SMS to 160 characters. Contact us if you need this setting enabled or disabled.</p> <p>The text needs to be URL encoded – Please refer to <b>Appendix B</b> for more information about URL encoding.</p> <p>We support Unicode characters. Eg: Chinese characters, smiley faces, emojis etc</p> <p></p> <p>As soon as you put one Unicode character in your message, the character count drops to 70 characters for a single message, then once you exceed one sms, the character count is 63 characters per segment. As per above, this tells the phone that it is part 1 of 2, part 2 of 2 etc.</p> <p>Unicode needs to be enabled on your account. Contact SMS Everyone for this feature.</p>
--	--	---

## Parameters for Scheduling a Campaign for a later date/time

<b>campaignid</b> campaign ref	No	<p>The unique identifier for your SMS campaign. Valid characters are Alphanumeric characters and dashes (-) only.</p> <p><b>A campaign Id of 0 (zero) is not permitted.</b></p> <p>This is useful for combining and scheduling campaigns together on the one campaign ID. Plus this Campaign Id is required if you need the campaign to be deleted later via our 'Delete Campaign' API.</p> <p><u>Note:</u> If you schedule a message to be sent with a specific Campaign Id, then schedule a different message to a different phone number with the same campaign ID, the message and originator of the second message will be ignored and this second message's recipient/s will be added to the existing Campaign Id. Therefore, the original message &amp; Originator will be sent to both phones.</p>
<b>time</b> <b>send_time</b>	No	<p>The date and time you would like the SMS message to be sent. If you don't provide this field, or the Date/Time is in the past, the SMS will be sent immediately.</p> <p>Required format is YYYYMMDDHHMM For example, 12 January 2017 at 11.30 am would be: <code>time=201701121130</code></p> <p>As above, if you schedule a campaign for a specific time, then schedule a second campaign with the same campaign ID at a different time, the second campaign will be added to the first and sent at the time the first campaign was scheduled for.</p>

## Parameters for modifying a campaign

<b>action</b>	Yes	<p>To modify a scheduled campaign, send <code>action=modify</code></p> <p>You can modify the send time, the originator (from number) and the message text. Username, Password and Campaign Id fields are required for this action plus the field that you wish to modify.</p> <p>For example, to modify the send time, submit username, password, the campaign id and the new send time.</p>
---------------	-----	--

## Parameters for deleting a campaign

<b>action</b>	Yes	<p>To delete a scheduled campaign, send <code>action=delete</code></p> <p>Username, Password and Campaign Id fields are also required for this action.</p>
---------------	-----	--

## Parameters for pausing a campaign

<b>action</b>	Yes	<p>To pause a scheduled or running campaign, send <code>action=pause</code></p> <p>To resume a campaign: <code>action=pause</code></p> <p>Username, Password and Campaign Id fields are also required for this action.</p>
---------------	-----	--

## Parameters for checking status of a Campaign

<b>action</b>	Yes	To check the status of a scheduled campaign, send <code>action=status</code> . Username, Password and Campaign Id fields are also required for this action. The HTTP response from us will be a number. The meaning of these numbers is described below: 0 – Deleted 1 – Completed (sent) 2 – Scheduled to be sent (active) 3 – Running (ie sending now) 4 – Paused (from scheduled) 5 – Paused (from running) -1 – Campaign Id not found
---------------	-----	---

## Send Status of Message

After submitting the HTTP request to the SMS Everyone Gateway, you should expect to get back an HTTP response. The response code you receive will indicate the success or failure of the SMS message being sent. A response with the number zero (0) indicates that you have successfully submitted your message to SMS Everyone, while any other response indicates that the message failed to be submitted.

### *Successful HTTP Response Sample*

```
0
```

A failed response has the error code or the error code and a description of the error condition.

### *Failure HTTP Response Sample*

```
-300 Incorrect User or Password
```

There are a number of failure codes that can be returned, depending on the reason for the message failure. See Appendix B for a list of valid expected response codes and descriptions.

### Additional HTTP response options

We can respond with custom HTTP responses to match your existing code and additional fields to give you extra information about your post.

For example, if you need something like 'OK' instead of 0 for a successful post, contact us and we can change it our end in a few seconds.

Additional fields are:

### Credits Used

We can respond with the number of credits that were used by your post to help you with billing your end.

For example, you could be sending a double message to 3 mobiles, so we can respond with 0 for success plus the number of credits used, separated by a comma.

0,6

## Reference ID

We can send you back the SMS Everyone reference ID of the message. Eg:

0,1234567

## 2-way (Receiving reply messages on demand)

To retrieve reply (MO) messages from our web service, send either an HTTP POST or GET to:

**HTTPS (Recommended)**

<https://api.smseveryone.com/ws/replies>

**HTTP**

<http://api.smseveryone.com/ws/replies>

### Retrieving All Unretrieved Replies

Parameters	Required	Description
<b>username</b> user_name user usr name	Yes	Username provided by SMS Everyone for use of the HTTP Interface.  The username field is <u>not</u> case sensitive
<b>password</b> pass pwd	Yes	Password provided by SMS Everyone for use of the HTTP Interface  Case sensitive
<b>Test</b> sandbox testmode testing	No	Whilst you're in testing mode, add &test=1 to retrieve SMS replies but we won't flag them as retrieved. Remove this field once you have completed testing.
<b>Format</b>	No	If you don't specify a format field, we will pass the replies to you in csv format. See <a href="#">HTTP RESPONSE - CSV</a> below.  If you specify format=json, we will pass the replies to you in json format (see <a href="#">HTTP RESPONSE – JSON</a> )  If you specify format=xml, we pass the data to you in XML (see <a href="#">HTTP RESPONSE – XML</a> )

## Retrieving All Replies for a given period

The HTTP response will show all replies for your selected period regardless of whether they have been retrieved previously or not.

Parameters	Required	Description
<b>username</b> user_name user usr name	Yes	Username provided by SMS Everyone for use of the HTTP Interface. The username field is <u>not</u> case sensitive
<b>password</b> pass pwd	Yes	Password provided by SMS Everyone for use of the HTTP Interface Case sensitive
<b>days</b>	No	The number of days you want to check. Eg <code>&amp;days=7</code> will retrieve all replies in the last 7 days.
<b>startdate</b>	No	The start date of the time frame you wish to search on. Format is YYYYMMDDHHMM Eg: <code>&amp;startdate=202012251030</code>
<b>enddate</b>	No	The end date of the time frame you wish to search on. Format is YYYYMMDDHHMM Eg: <code>&amp;enddate=202012271030</code>
<b>origin</b>	No	The specific mobile number you wish to search for in international format without the + sign. Eg 61400111222 So for example, you could search for all replies from the number 61400123456 for the period from march 1 <sup>st</sup> 2020 to March 31 <sup>st</sup> 2020: <code>...&amp;origin=61400123456&amp;startdate=202003010000&amp;enddate=202003312400...</code>
<b>refid</b>	No	Values = 0 (off) or 1 (on) When <b>refid=1</b> we pass back a reference ID for each reply message in the response
<b>Format</b>	No	If you don't specify a format field, we will pass the replies to you in csv format. See <b>HTTP RESPONSE</b> below. If you specify <code>format=json</code> , we will pass the replies to you in json format If you specify <code>format=xml</code> , we pass the data to you in XML

## HTTP RESPONSE - CSV

Plain text is returned. We only show the replies that we have received since you last checked. Ie whenever you poll for replies, we mark those reply messages as retrieved in our database so that we don't show them to you again.

The HTTP response CSV format is:

ResponseCode,#OfMessages,Mobile1,Message1,DateTime1,Mobile2,Message2,DateTime2,etc...

Example:

```
0,2,0402756333,Yes,2014-11-12 01:32,0439897906,Yes,2014-11-12 01:32
```

If there are no messages to be retrieved:

```
0,0
```

## Notes:

- ResponseCode will always be 0 (zero) unless there is an error.
- #OfMessages = the number of replies we are passing to you in this HTTP response
- DateTime is in the following format: YYYY-MM-DD HH:mm (24 hour format)
- Mobile is in 04 format for all Australian mobiles and International format for all international mobiles
- Standard CSV formatting applies – For example:
  - Commas within the text will result in speech marks either side of the text so you know this particular comma is not a delimiter. Eg:  
**Yes, I'll be there** will appear as "Yes, I'll be there"
  - Speech marks will result in double speech marks. Eg:  
**I'm going to be "late" today** will appear as "I'm going to be ""late"" today"

## HTTP RESPONSE – JSON

- If there are no messages, response will be:

```
{  
  "Count": 0,  
  "Messages": []  
}
```

- Count = the number of replies we are passing to you in this response
- ‘Received’ date time is in this format: YYYY-MM-DD HH:mm:ss (24 hour format)
- All mobile numbers are in international format without the + sign
- We include a reference ID for each reply message
- Example:

```
{  
  "Count": 1,  
  "Messages": [  
    {  
      "Received": "2020-06-04 16:39:14",  
      "Originator": "61402756333",  
      "MessageText": "赶来",  
      "ReferenceId": "4436768"  
    }  
  ]  
}
```

## HTTP RESPONSE – XML

- If there are no messages, response will be:

```
<Replies>  
  <Count>0</Count>  
  <Messages />  
</Replies>
```

- Count = the number of replies we are passing to you in this response
- ‘Received’ date time is in this format: YYYY-MM-DD HH:mm:ss (24 hour format)

- All mobile numbers are in international format without the + sign
- We include a reference ID for each reply message
- Example:

```
<Replies>
  <Count>1</Count>
  <Messages>
    <Reply>
      <Received>2020-06-04 16:40:34</Received>
      <Originator>61402756333</Originator>
      <MessageText>, "</MessageText>
      <ReferenceId>4436771</ReferenceId>
    </Reply>
  </Messages>
</Replies>
```

## Checking SMS Credit Balance

To check your balance of SMS credits, send an HTTP POST or GET to

**HTTPS (Recommended)**

<https://api.smseveryone.com/ws/creditcheck>

**HTTP**

<http://api.smseveryone.com/ws/creditcheck>

Parameters	Required	Description
<b>username</b> user user_name usr name	Yes	Username provided by SMS Everyone for use of the HTTP Interface. The username field is <u>not</u> case sensitive
<b>password</b> pass pwd	Yes	Password provided by SMS Everyone for use of the HTTP Interface Case sensitive

The HTTP response is plain text showing only of the number of credits.

Example:

203

A zero balance will show

0

## Add Credits

You can add credits to an account via our **add credits** API call. The usage scenario is more for software partners who have many sub-clients. Your software can hit our API requesting that the sub-account be topped up by x number of credits.

### SOME EXAMPLES:

1. MANUAL TOP UP BY SUB-CLIENT – In your software have a credits top-up section that allows your client to select the number of credits that they want to top up by (eg: 1000, 2000, 5000 & 10,0000) or type in the number of credits manually.

Your software then hits our API which tops them up immediately and we respond with the new balance so you know that it worked. Then make sure that your software sends you an alert of some kind so that you can bill your client. We then bill you at the end of the month for all top ups in the previous month.

2. AUTOMATIC TOP UP – Your software detects whenever the account is getting low by hitting the above **creditcheck** API before every send. When the balance is below a certain threshold, hit our **add credits** API and top them up automatically.

**Please note: This is only for pre-approved clients and will return an error if you attempt to add credits via this method without approval.**

Below are the API calls pertaining to adding credits:

**HTTPS (Recommended)**

<https://api.smseveryone.com/ws/credits>

**HTTP**

<http://api.smseveryone.com/ws/credits>

Parameter	Required	Description
<b>action</b>	Yes	To add credits, you'll need action=add
<b>credits</b>	Yes	This is the number of credits you wish to add. Eg credits=1000. A negative number is not possible.
<b>Username</b>	Yes	The username of the sub-account.
<b>Password</b>	Yes	The password of the sub-account

The HTTP response is plain text showing the standard "0" response plus the new balance of the account.

For example, this is a successful top up with the new balance being 1023 credits:

```
0,1023
```

Trying to add a negative number of credits error:

```
-130,Invalid amount of credits to add
```

Trying to add credits to an account that does not have approval:

```
-129,Not allowed to add credits
```

## Optouts

SMS everyone maintains an optout database for every client. We wash every single sms sent against your optout database. Anybody who is opted out cannot be sent an SMS.

If you are using our API to send marketing messages, you will need to provide either your own optout functionality your end or you can use our optout system. Either way you will probably need an optout number so that your end recipients can reply to opt out. Contact us for an optout number if required.

Below are the API calls pertaining to opt outs:

**HTTPS (Recommended)**

<https://api.smseveryone.com/ws/optouts>

**HTTP**

<http://api.smseveryone.com/ws/optouts>

### Add a number to our optout database (opt out / unsubscribe)

<b>action</b>	Yes	To add a number, youll need action=add
<b>origin</b>	Yes	This is the mobile number you wish to add. Eg origin=61400111222
<b>Username</b>	Yes	As per the campaign API, the username is required
<b>Password</b>	Yes	Also required

### Delete a number from our optout database (opt back in / re-subscribe)

By deleting a number from our optout database, you acknowledge that the recipient has opted back in.

<b>action</b>	Yes	To delete a number (opt back in), youll need action=delete
<b>origin</b>	Yes	This is the mobile number you wish to delete. Eg origin=61400111222
<b>Username</b>	Yes	As per the campaign API, the username is required
<b>Password</b>	Yes	Also required

## Retrieve a list of all numbers that are opted out / unsubscribed

<b>action</b>	Yes	To retrieve a list of all numbers, you'll need action=list
<b>Username</b>	Yes	As per the campaign API, the username is required
<b>Password</b>	Yes	Also required
<b>Format</b>	No	If you don't specify the format, we give the data to you in plain text CSV format Format=json – returns data in json format Format=xml – returns data in XML format

## Extras

### Time zones

We can set your account to any time zone in the world so that when you specify the send time, it will be in that countries' time zone. Contact SMS Everyone to set this up.

### Send Window

To prevent your staff or clients sending sms messages at inconvenient times, we can enable a send window on your account. This prevents sending sms outside of whatever hours you specify.

If you schedule/send before the window opens, we queue it until the window opens.

If you schedule/send after the window closes, we queue it for the following day when the window opens.

Contact SMS Everyone to set this up.

## Samples

### Sample for Sending a Text Message immediately

```
https://api.smseveryone.com/ws/campaign?user=UserName&password=Password&originator=JIMSBAR&destination=61412333444&message>Hello_There!
```

### Sample for Scheduling a Text Message to be sent later

```
https://api.smseveryone.com/ws/campaign?username=UserName&password=Password&originator=JIMSBAR&destination=61412333444&message>Hello_There!&time=201501010900&campaignid=123
```

### Sample for modifying a scheduled Campaign

#### changing originator

```
https://api.smseveryone.com/ws/campaign?username=UserName&password=Password&campaignid=123&action=modify&originator=changed
```

#### changing send time

```
https://api.smseveryone.com/ws/campaign?username=UserName&password=Password&campaignid=123&action=modify&send_time=202003061530
```

#### changing message text

```
https://api.smseveryone.com/ws/campaign?username=UserName&password=Password&campaignid=123&action=modify&message_text=changed
```

### Sample for deleting a scheduled Campaign

```
https://api.smseveryone.com/ws/campaign?username=UserName&password=Password&campaignid=123&action=delete
```

### **Sample for checking the status of a scheduled Campaign**

`https://api.smseveryone.com/ws/campaign?username=UserName&password=Password&campaignid=123&action=status`

### **Sample for checking for all unretrieved replies**

`https://api.smseveryone.com/ws/replies?username=UserName&password=Password`

### **Sample for checking for all replies in the last 7 days**

`https://api.smseveryone.com/ws/replies?username=UserName&password=Password&days=7`

### **Sample for checking for all replies from April 1<sup>st</sup> 2020 – April 3<sup>rd</sup> 2020 from 6402111222**

`https://api.smseveryone.com/ws/replies?username=UserName&password=Password&startdate=202004010000&enddate=202004032400&origin=6402111222`

## **Replies Samples**

### **HTTP GET String Sent from SMS Everyone to Your Site**

`http://yourwebsite.aspx?username=xyz&password=xyz&originator=61402756334&recipient=61429848254&reference=4255525&message>Hello%20There`

## **Optouts Samples**

### **Opt a number out**

`https://api.smseveryone.com/ws/optouts?user=Username&password=Password&action=add&origin=61400222333`

### **Opt a number back in**

`https://api.smseveryone.com/ws/optouts?user=Username&password=Password&action=delete&origin=61400222333`

### **Retrieve a list of all opt outs**

`https://api.smseveryone.com/ws/optouts?user=Username&password=Password&action=list`

The plain text CSV HTTP response will be in the following format:

Mobile number 1,Date opted out,Message text (if available),How they opted out (originator = the end user opted out, Client = the client manually opted them out),Mobile number 2,etc

For example

`61400555555,05-12-2018 07:52,Stop please,Originator,61400999999,02-12-2016 04:11,,Client`

JSON format:

```
{  
    "Count": 4,  
    "OptOut": [  
        {  
            "Added": "2020-06-09 11:44:24",  
            "Originator": "61400000000",  
            "MessageText": "STOP",  
            "Reason": "Originator"  
        },  
        {  
            "Added": "2020-06-09 11:44:24",  
            "Originator": "61400000001",  
            "MessageText": "",  
            "Reason": "Client"  
        },  
        {  
            "Added": "2020-06-09 11:42:03",  
            "Originator": "61400999777",  
            "MessageText": "",  
            "Reason": "Client"  
        },  
        {  
            "Added": "2018-12-05 18:52:42",  
            "Originator": "61400555555",  
            "MessageText": "",  
            "Reason": "Client"  
        }  
    ]  
}
```

XML format:

```
<OptOuts>  
<Count>4</Count>  
<OptOut>  
<OptOut>  
<Added>2020-06-09 11:44:24</Added>  
<Originator>61400000000</Originator>  
<MessageText/>STOP<MessageText/>  
<Reason>Originator</Reason>  
</OptOut>  
<OptOut>  
<Added>2020-06-09 11:44:24</Added>  
<Originator>61400000001</Originator>  
<MessageText/>  
<Reason>Client</Reason>  
</OptOut>  
<OptOut>  
<Added>2020-06-09 11:42:03</Added>  
<Originator>61400999777</Originator>  
<MessageText/>  
<Reason>Client</Reason>  
</OptOut>  
<OptOut>  
<Added>2018-12-05 18:52:42</Added>  
<Originator>61400555555</Originator>  
<MessageText/>  
<Reason>Client</Reason>  
</OptOut>  
</OptOut>  
</OptOuts>
```

# Appendices

## Appendix A: Http Response Codes

Error code	Error description
-506	Gateway offline. Unknown reason
-505	Gateway offline. Manually set
-504	Gateway offline. End-point unavailable
-503	Default gateway offline. No failover gateways found. Using system default gateway
-502	Default gateway offline. Campaign processing halted as using failover gateways not allowed
-501	Default gateway offline. Using system default gateway
-500	Default gateway offline. Using failover gateways
-207	Credit record already exists for client
-206	Credit account is inactive
-205	Insufficient business grade credits available to complete campaign
-204	Insufficient business grade credits available to start campaign
-203	Insufficient credits available to complete campaign
-202	Insufficient credits available to start campaign
-201	Credits to use for campaign not specified
-200	Credits remaining enquiry failed to locate Credit record
-130	Invalid amount of credits to add
-129	Not allowed to add credits
-128	CrmId not found
-127	CampaignId not found
-125	No valid Destinations or ListIds found to schedule
-124	Modification request after campaign start time cutoff
-123	Attempt to retrieve status of non-existent campaign. Check Campaign Id
-122	Invalid action specified
-121	Attempt to remove numbers from invalid campaign
-120	Attempt to modify a non-existent campaign. Check Campaign Id
-119	Campaign Unicode characters not allowed in Message Text
-118	Campaign ID not specified for status action

-117	Campaign message text not specified
-116	Campaign message length too long. Specify concat option for longer messages
-115	Campaign schedule date invalid
-114	Campaign originator contains invalid characters
-113	Campaign originator length invalid
-112	Campaign not specified for delete action
-111	Campaign no destination number(s) specified
-110	Campaign start time has already past
-109	Campaign rescheduled due to unknown reason
-108	Campaign rescheduled due to outage
-107	Deletion request for previously deleted campaign
-106	Deletion request for completed campaign
-105	Campaign deletion request for campaign that does not exist
-104	Deletion request after campaign start time cutoff
-103	Campaign create attempted to modify a completed campaign
-102	Campaign create attempted to modify a running campaign
-101	Campaign create attempted to modify a deleted campaign
-100	Campaign create attempt with no CampaignID or ClientRef specified
0	Message is OK (so far). Or successful response.

## Appendix B: URL Encoding

### Why do we need URL Encoding?

The specification for URLs poses a problem, in that it limits the use of allowed characters in URLs to only a limited subset of the US-ASCII character set. This means that messages using characters not contained within that character set must be represented by alternative methods.

### How are characters URL Encoded?

URL encoding of a character consists of a "%" symbol, followed by the two-digit hexadecimal representation (case-insensitive) of the ISO-Latin code point for the character.

#### Example

Space = decimal code point 32 in the ISO-Latin set.

32 decimal = 20 in hexadecimal

The URL encoded representation will be %20

Line feed = %0A

### What characters need to be encoded and why?

- **Reserved Characters**

**WHY:** These characters have a special role or syntax within URL's and should not be used as part of a parameter value. These characters must always be encoded.

#### WHICH CHARACTERS:

Character	Code Points (Hex)	Code Points (Decimal)
Dollar ("\$")	24	36
Ampersand ("&")	26	38
Plus ("+")	2B	43
Comma (",")	2C	44
Forward slash/Virgule ("/")	2F	47
Colon (":")	3A	58
Semi-colon (";")	3B	59
Equals ("=")	3D	61
Question mark ("?")	3F	63
'At' symbol ("@")	40	64

- **Un-Safe Characters**

**WHY:** Some characters present the *possibility* of being misunderstood within URLs for various reasons. These characters should also always be encoded

**WHICH CHARACTERS:**

Character	Code Points (Hex)	Code Points (Dec)	Why encode?
Space	20	32	Significant sequences of spaces may be lost in some uses (especially multiple spaces)
'Quotation marks' 'Less Than' symbol ("<") 'Greater Than' symbol (">")	22 3C 3E	34 60 62	These characters are often used to delimit URLs in plain text.
'Pound' or 'Hash' character ("#")	23	35	This is used in URLs to indicate where a fragment identifier (bookmarks/anchors in HTML) begins.
Percent character ("%")	25	37	This is used to URL encode/escape other characters, so it should itself also be encoded.

- BLOCKED CHARACTERS**

The characters below are not permitted to be sent via SMS. If you send these characters via our API, we will replace them with a space.

- Left Curly Brace {
- Right Curly Brace }
- Vertical Bar/Pipe |
- Backslash \
- Caret ^
- Tilde ~
- Left Square Bracket [
- Right Square Bracket ]
- Grave Accent `

- ASCII Control Characters**

**Why:** These characters are not printable and should not be included in Text messages although some newer phones allow these characters to be used and sent as messages.

**WHICH CHARACTERS:** (Carriage Return, Page Break etc.)

- Non-ASCII Characters**

**WHY:** These are not, by definition, legal in URL's as they are not part of the ASCII Character Set