



MyMobileAPI  
[mymobileapi.com](http://mymobileapi.com)

---

Web Service API Guide

---

# TABLE OF CONTENTS

Overview .....	3
Checking Credits .....	4
Sending Your Data .....	5
Sending Your Data - (To a Group Online).....	10
Retrieving Sent Data - (Overview).....	14
So how is this achieved? .....	14
Some tips:.....	14
Limitations:.....	14
Retrieving Sent Data - (Dataset and XML).....	15
Retrieving Replies - (Overview).....	17
So how is this achieved? .....	17
Some tips:.....	17
Limitations:.....	17
Retrieving Replies - (Dataset and XML).....	18
Getting Your Groups.....	20
Retrieving - Your Short Code Counts .....	22



# OVERVIEW

The web service can be referenced at <http://www.mymobileapi.com/api5/api.asmx>

The naming convention describes what the function accepts and returns. EG: Reply\_STR\_DS will accept an XML string and return a Dataset (.net object).

Most of the functions follow a common thread: Accept 3 parameters, return XML or dataset.  
The returned data will always contain:

```
</api_result>  
  <call_result>  
    <result>True</result>  
    <error />  
  </call_result>  
</api_result>
```

The result will either be "True" or "False". If "False" the error will be displayed with the "error" tag.

All entries marked in BLUE below are optional parameters.

# CHECKING CREDITS

There are 2 methods to check your credits:

- Credits\_STR (Returns XML string)
- Credits\_DS (Returns Dataset)

Each function accepts 2 parameters:

- Username (String)
- Password (String)

The return object looks like:

```
<api_result>
  <data>
    <credits>xxxxxx</credits>
  </data>
  <call_result>
    <result>True</result>
    <error />
  </call_result>
</api_result>
```

Your current credit value will be displayed between the “credits” tag.

# SENDING YOUR DATA

To send is relatively easy, however due to the options it can become confusing.

In a nutshell:

If you specify a default then it will only be utilized if the entry data does not exist. If both don't exist then it will throw an error.

There are 5 methods to send your messages:

- Send\_DS\_DS (Accepts Dataset, returns Dataset)
- Send\_STR\_DS (Accepts XML string, returns Dataset)
- Send\_STR\_STR (Accepts (XML string, returns XML String)
- Send\_DS\_STR (Accepts Dataset, returns XML String)
- Send\_ZIP\_ZIP (Accepts zipped stream, returns zipped stream)

Each function accepts 3 parameters

- Username (String)
- Password (String)
- XML String or Dataset or compressed byte array (depending on the function you choose)

The XML objects looks like

```
<senddata>
  <settings>
    <live>True</live>
    <return_credits>True</return_credits>
    <return_msgs_credits_used>True</return_msgs_credits_used>
    <return_msgs_success_count>True</return_msgs_success_count>
    <return_msgs_failed_count>True</return_msgs_failed_count>
    <return_entries_success_status>True</return_entries_success_status>
    <return_entries_failed_status>True</return_entries_failed_status>
    <default_senderid />
    <default_date>04/Mar/2009</default_date>
    <default_time>11:15</default_time>
    <default_curdate>04/Mar/2009</default_curdate>
    <default_curtime>11:15</default_curtime>
    <default_data1>This is a default msg</default_data1>
    <default_data2 />
    <default_flash>False</default_flash>
    <default_type>SMS</default_type>
    .....
  </settings>
</senddata>
```

```
.....
  <default_costcentre>NA</default_costcentre>
  <default_validityperiod>0</default_validityperiod>
</settings>
<entries>
  <numto>0832297941</numto>
  <customerid>UnqieValue1</customerid>
  <senderid>Test me</senderid>
  <time>11:15</time>
  <data1>This is a test message 1!</data1>
  <data2 />
  <flash>False</flash>
  <type>SMS</type>
  <costcentre>NA</costcentre>
  <validityperiod>0</validityperiod>
</entries>
<entries>
  <numto>0832297941</numto>
  <customerid>UnqieValue1</customerid>
  <senderid>Test me</senderid>
  <time>11:15</time>
  <data1 />
  <data2 />
  <flash>False</flash>
  <type>SMS</type>
  <costcentre>NA</costcentre>
  <validityperiod>0</validityperiod>
</entries>
<entries>
  <numto>0832297941</numto>
  <customerid>c60f9f17e3084ae187ca0bb36691c75f</customerid>
  <senderid>Test me</senderid>
  <time>11:15</time>
  <data1>James Pearce</data1>
  <data2>083 229 7941</data2>
  <flash>False</flash>
  <type>VCARD</type>
  <costcentre>NA</costcentre>
  <validityperiod>0</validityperiod>
</entries>
<entries>
  <numto>0832297941</numto>
  <customerid>5d488bc70fb14476af3d46dcf867c02f</customerid>
  <senderid>Test me</senderid>
  <time>11:15</time>
  <data1>Click here for MyMobileAPI</data1>
  <data2>http://www.mymobileapi.com</data2>
  <flash>False</flash>
  <type>WPUSH</type>
  <costcentre>NA</costcentre>
  <validityperiod>0</validityperiod>
</entries>
</senddata>
```

The following determines how the result is formatted as well as defines the defaults.

XML Tag	Description	Default
live	If Live = false then it will return the result but won't send the data. It won't appear in your sent report.	True
return_credits	If True then the current credits count will be returned.	False
return_msgs_credits_used	If True then the amount of credits used will be returned	False
return_msgs_success_count	If True then the total success data count will appear in the result.	False
return_entries_failed_status	If True then the total failed data count will appear in the result.	False
return_entries_success_status	If True then the data that is accepted will be returned in the result.	False
return_entries_failed_status	If True then the data that is rejected will be returned in the result.	False
default_senderid	This is the default setting for the call but can be overridden by the individual entries. Only certain accounts can alter this.	Repliable
*default_date	This is the date the messages must be sent out. Format is dd/MMM/yyyy eg: 13/apr/2009	-
*default_time	This is the time the messages must be sent out. Format is HH:mm eg: 17:34	-
default_curdate	This is the current date on your server. It is used to determine when your messages should go out. If not used then the server date is used.	Server Date
default_curtime	This is the current time on your server. It is used to determine when your messages should go out. If not used then the server time is used.	Server Date
default_data1	This is the default setting for the call but can be overridden by the individual entries.	""
default_data2	This is the default setting for the call but can be overridden by the individual entries.	""
default_flash	This is the default setting for the call but can be overridden by the individual entries.	False
default_type	This is the default setting for the call but can be overridden by the individual entries. Values can be SMS / VCARD / WPPUSH / PORT	SMS
default_costcentre	Will be used in future.	""
mo_forwardemail	All incoming replies will be forwarded to this email address	""
default_validityperiod	The amount of time in hours an SMS should remain valid. The network will continue to try to deliver the SMS over the validity period.	-

(All entries marked with \* are Mandatory)

For each Entry in the XML you require the following can be utilized.

XML tag	Description	Default
numto	The number the message must go to.	SMS
*customerid	The customerID is a field that is populated that has meaning to your system. This field can be returned in the result sets and can make updating data very easy. We suggest using a GUID or Identity key.	""
senderid	The senderid can be altered (this is the number the message is received from). Only certain accounts allow this feature.	""
time	The time the message must be delivered. This is joined with the default_date. See default_time for formatting.	-
data1	This is data that will be sent	""
data2	This may be required depending on the TYPE of message you wish to send	""
flash	IF set to TRUE the message will display immediately on the users screen. It however does not get stored in the users inbox.	False
type	This is the default setting for the call but can be overridden by the individual entries. Values can be SMS / VCARD / WPPUSH / PORT	SMS
costcentre	Will be used in future.	""
validityperiod	The amount of time in hours an SMS should remain valid. The network will continue to try to deliver the SMS over the validity period. This value will override the 'default_validityperiod' setting.	-

(All entries marked with \* are Mandatory)

XML tag	Description
entries_failed	<p>If this tag exists then some of the data you pushed has errors. It will contain: "numto", "customerid" and "reason". The reason will be</p> <ul style="list-style-type: none"> <li>• numto invalid</li> <li>• senderid invalid</li> <li>• time invalid</li> <li>• flash invalid</li> <li>• costcentre invalid</li> <li>• type invalid</li> <li>• data1 invalid</li> <li>• data2 invalid</li> <li>• data1 + data2 invalid</li> </ul>
entries_success	If this tag exists then it will contain "numto" and "customerid".
send_info	<p>This describes the overall data returned. It WILL contain "eventid" and COULD have "credits", "msgs_credits_used", "msgs_success" and "msgs_failed".</p> <ul style="list-style-type: none"> <li>• Credits = credits remaining in account</li> <li>• Msgs_credits_used = the amount of credits used</li> <li>• Msgs_success = a count of the messages accepted for delivery</li> <li>• Msgs_failed = a count of the messages rejected</li> </ul>



The output of the following will be defined by the input of the above parameters:

```
<api_result>
  <entries_failed>
    <numto />
    <customerid />
    <reason>numto invalid</reason>
  </entries_failed>
  <entries_success>
    <numto>27832297941</numto>
    <customerid>UnqieValue1</customerid>
  </entries_success>
  <entries_success>
    <numto>27832297941</numto>
    <customerid>UnqieValue1</customerid>
  </entries_success>
  <entries_success>
    <numto>27832297941</numto>
    <customerid>d47e1efd□5e4f□4832□ba7f□07f35d8c10f9</customerid>
  </entries_success>
  <send_info>
    <eventid>27105067</eventid>
    <credits>39763</credits>
    <msgs_credits_used>3</msgs_credits_used>
    <msgs_success>3</msgs_success>
    <msgs_failed>1</msgs_failed>
  </send_info>
  <call_result>
    <result>True</result>
    <error />
  </call_result>
</api_result>
```

## SENDING YOUR DATA - (TO A GROUP ONLINE)

To send to a group online you are required to pass in the GROUPID for the specific group you wish to target. This can be attained from a function within the API.

There are 4 methods to send your messages:

- Send\_DS\_DS (Accepts Dataset, returns Dataset)
- Send\_STR\_DS (Accepts XML string, returns Dataset)
- Send\_STR\_STR (Accepts XML string, returns XML String)
- Send\_DS\_STR (Accepts Dataset, returns XML String)
- Send\_ZIP\_ZIP (Accepts zipped stream, returns zipped stream)

Each function accepts 3 parameters

- Username (String)
- Password (String)
- XML String or Dataset or compressed byte array (depending on the function you choose)

The XML object looks like:

```
<senddata>
  <settings>
    <live>True</live>
    <return_credits>True</return_credits>
    <return_msgs_credits_used>True</return_msgs_credits_used>
    <return_msgs_success_count>True</return_msgs_success_count>
    <return_msgs_failed_count>True</return_msgs_failed_count>
    <return_entries_success_status>True</return_entries_success_status>
    <return_entries_failed_status>True</return_entries_failed_status>
    <default_senderid />
    <default_date>04/Mar/2009</default_date>
    <default_time>11:15</default_time>
    <default_curdate>04/Mar/2009</default_curdate>
    <default_curtime>11:15</default_curtime>
    <default_data1>This is a default msg</default_data1>
    <default_data2 />
    <default_flash>False</default_flash>
    <default_type>SMS</default_type>
    <default_costcentre>NA</default_costcentre>
    <default_validityperiod>0</default_validityperiod>
    <groupid>NA</groupid>
  </settings>
</senddata>
```

The following determines how the result is formatted as well as defines the defaults.

XML tag	Description	Default
live	If Live = false then it will return the result but won't send the data. It won't appear in your sent report.	True
return_msgs_credits_used	If True then the amount of credits used will be returned.	False
return_credits	If True then the current Credits count will be returned.	False
return_msgs_success_count	If True then the total success data count will appear in the result.	False
return_entries_failed_status	If True then the total failed data count will appear in the result.	False
return_entries_success_status	If True then the data that is accepted will be returned in the result.	False
return_entries_failed_status	If True then the data that is rejected will be returned in the result.	False
default_senderid	This is the default setting for the call but can be overridden by the individual entries. Only certain accounts can alter this.	Repliable
*default_date	This is the date the messages must be sent out. Format is dd/MMM/yyyy eg: 13/apr/2009	-
*default_time	This is the time the messages must be sent out. Format is HH:mm eg: 17:34	-
default_curdate	This is the current time on your server. It is used to determine when your messages should go out. If not used then the server time is used.	Server Date
default_curttime	This is the current time on your server. It is used to determine when your messages should go out. If not used then the server time is used.	Server Time
default_data1	This is the default setting for the call but can be overridden by the individual entries.	""
default_data2	This is the default setting for the call but can be overridden by the individual entries.	""
default_flash	This is the default setting for the call but can be overridden by the individual entries.	False
default_type	This is the default setting for the call but can be overridden by the individual entries. Values can be SMS / VCARD / WPPUSH / PORT	SMS
default_costcentre	Will be used in future.	""
mo_forwardemail	All incoming replies will be forwarded to this email	""
default_validityperiod	The amount of time in hours an SMS should remain valid. The network will continue to try to deliver the SMS over the validity period.	-
*groupid	The groupid to send to	0

(All entries marked with \* are Mandatory)

The output of the following will be defined by the input of the above parameters:

```
<api_result>
  <entries_failed>
    <numto />
    <customerid />
    <reason>numto invalid</reason>
  </entries_failed>
  <entries_success>
    <numto>27832297941</numto>
    <customerid />
  </entries_success>
  <entries_success>
    <numto>27832297941</numto>
    <customerid />
  </entries_success>
  <entries_success>
    <numto>27832297941</numto>
    <customerid />
  </entries_success>
  <send_info>
    <eventid>27105067</eventid>
    <credits>39763</credits>
    <msgs_credits_used>3</msgs_credits_used>
    <msgs_success>3</msgs_success>
    <msgs_failed>1</msgs_failed>
  </send_info>
  <call_result>
    <result>True</result>
    <error />
  </call_result>
</api_result>
```

XML tag	Description
entries_failed	<p>If this tag exists then some of the data you pushed has errors. It will contain: "numto", "customerid" and "reason". The reason will be</p> <ul style="list-style-type: none"> <li>• numto invalid</li> <li>• senderid invalid</li> <li>• time invalid</li> <li>• flash invalid</li> <li>• costcentre invalid</li> <li>• type invalid</li> <li>• data1 invalid</li> <li>• data2 invalid</li> <li>• data1 + data2 invalid</li> </ul>
entries_success	<p>If this tag exists then it will contain "numto" and "customerid".</p>
send_info	<p>This describes the overall data returned. It WILL contain "eventid" and COULD have "credits", "msgs_credits_used", "msgs_success" and "msgs_failed".</p> <ul style="list-style-type: none"> <li>• Credits = credits remaining in account</li> <li>• msgs_credits_used = the amount of credits used</li> <li>• Msgs_success = a count of the messages accepted for delivery</li> <li>• Msgs_failed = a count of the messages rejected</li> </ul>

Due to the input you cannot specify a CustomerID per number if sending to a group the CustomerID will be blank.

# RETRIEVING SENT DATA - (OVERVIEW)

The SMS Gateway is designed to allow easy retrieval of your sent items.

The design allows the following:

- Retrieve items you have not already pulled
- Instant retrieval of data (due to API table design and indexing)
- Customizable result formatting
- Prevents timeouts

## So how is this achieved?

The API system uses a unique identity field called “ChangeID” which is permanently incrementing. When a message status is received from the corresponding network a new record is written to the API and hence the ChangeID will increment by 1. Messages will not be retrieved via the API that do not have any status changes.

In the result set the element “changeid” will be present along with the other elements you have specified. This needs to be stored and the maximum changeid utilized by the next call. The following is occurring with the SMS Gateway:

```
“Select top 100 * from API_Sent_Table where changeid > @ID order by changeid asc”
```

## Some tips:

Start with id (changeid) = 0

Store the max changeid in your local system permanently (database or IO system)  
Only pull back data you require using the “cols\_returned” element, this will reduce bandwidth

## Limitations:

Sent items are only retained up to seven days, i.e. sent items that were sent more than a week ago cannot be retrieved via Web Services.

Only messages that were sent via Web Services, and from the account that was used to authenticated with, can be retrieved. E.g. sub accounts cannot retrieve each other’s messages and messages sent via FTP cannot be retrieved via Web Services.

## RETRIEVING SENT DATA - (DATASET AND XML)

Getting sent data from the system:

There are 4 methods to retrieve sent items:

- Sent\_DS\_DS (Accepts Dataset, returns Dataset)
- Sent\_STR\_DS (Accepts XML string, returns Dataset)
- Sent\_STR\_STR (Accepts XML string, returns XML string)
- Sent\_DS\_STR (Accepts Dataset, returns XML string)
- Sent\_ZIP\_ZIP (Accepts zipped stream, returns zipped stream)

Each function accepts 3 parameters

- Username (String)
- Password (String)
- XML String or Dataset (depending on the function you choose)

The XML object looks like:

```
<sent>
  <settings>
    <id>0</id>
    <max_recs></max_recs>
    <cols_returned> </cols_returned>
    <date_format>yyyyMMddHHmmss</date_format>
  </settings>
</sent>
```

Parameter	Description	Default
*Id	The max CHANGEID from the previous call. Start with "0"	-
Max_recs	The number of records returned from the call. Values 1 to 100	100
Cols_Returned	The xml tags returned in the result. Options are : sentid,eventid, smstype,numto,data,flash,customerid,status,statusdate. Each entry must be separated with a comma.	ChangeID
Date Format	The format of any dates that are returned.	dd/MMM/yyyy HH:mm:ss

(All entries marked with \* are Mandatory)

The output returned will be a dataset or XML in the following format:

```

<api_result>
  <data>
    <changeid></changeid>
    <sentid></sentid>
    <eventid></eventid>
    <smstype> </smstype>
    <numto></numto>
    <data></data>
    <flash> </flash>
    <customerid> </customerid>
    <status> </status>
    <statusdate></statusdate>
  </data>
  <call_result>
    <result></result>
    <error />
  </call_result>
</api_result>
    
```

XML tag	Description
changeid	A unique ID identifying the change in a message status
SentID	A unique value for each message. You could receive the same SentID back if the message status changes.
eventid	When sending an eventid is generated and returned in the result xml. This eventid can be for 1 or "infinity" message entries.
smstype	This is the type of message sent. Options are "SMS", "VCARD", "WPUSH" or "PORT"
numto	The number the message was sent to
data	The data sent to the user
flash	If the message was a flash message
customerid	The customerid for the message when the data was sent
status	The message status. Options are "DELIVRD", "UNDELIV", "UNKNOWN", "EXPIRED"
statusdate	The date and time the message was sent



# RETRIEVING REPLIES - (OVERVIEW)

The SMS Gateway is designed to allow easy retrieval of your inbound messages (replies).

The design allows the following:

- Retrieve items you have not already pulled
- Instant retrieval of data (due to API table design and indexing)
- Customizable result formatting
- Prevents timeouts

## So how is this achieved?

The API system uses a unique identity field called “ReplyID” which is permanently incrementing. When an inbound message is received from the corresponding network a new record is written to the API and hence the ReplyID will increment by 1.

In the result set the element “ReplyID” will be present along with the other elements you have specified. This needs to be stored and the maximum replyid utilized by the next call. The following is occurring with the SMS Gateway:

```
“Select top 100 * from API_Reply_Table where replyid > @ID order by replyid asc”
```

## Some tips:

Start with id (replyid) = 0

Store the max replyid in your local system permanently (database or IO system)

Only pull back data you require using the “cols\_returned” element, this will reduce bandwidth

## Limitations:

Inbound messages are only retained for up to seven days, i.e. replies that were sent more than a week ago cannot be retrieved via Web Services.

Only replies to messages that were sent via Web Services, and from the account that was used to authenticate with, can be retrieved. E.g. sub accounts cannot retrieve each other’s messages and replies to messages sent via FTP cannot be retrieved via Web Services.

# RETRIEVING REPLIES - (DATASET AND XML)

There are 4 methods to retrieve the replies:

- Reply\_DS\_DS (Accepts Dataset, returns Dataset)
- Reply\_STR\_DS (Accepts XML string, returns Dataset)
- Reply\_STR\_STR (Accepts XML string, returns XML String)
- Reply\_DS\_STR (Accepts Dataset, returns XML String)
- Reply\_ZIP\_ZIP (Accepts zipped stream, returns zipped stream)

Each function accepts 3 parameters

- Username (String)
- Password (String)
- XML String or Dataset (depending on the function you choose)

The XML object looks like:

```
<reply>
  <settings>
    <id></id>
    <max_recs></max_recs>
    <cols_returned> </cols_returned>
    <date_format> </date_format>
  </settings>
</reply>
```

Parameter	Description	Default
*Id	The max REPLYID from the previous call	-
Max_recs	The number of records returned from the call. Values 1 to 100	100
Cols_Returned	The xml tags returned in the result. Options are: replyid, eventid, numfrom, receiveddata, received, sentid, sentdata, sentdatetime, sentcustomerid, optout Each entry must be separated with a comma.	Replyid
Date Format	The format of any dates that are returned.	dd/MMM/yyyy HH:mm:ss

(All entries marked with \* are Mandatory)

The output returned will be a dataset or XML in the following format:

```

<api_result>
  <data>
    <replyid></replyid>
    <eventid></eventid>
    <numfrom></numfrom>
    <receiveddata> 123</receiveddata>
    <sentid></sentid>
    <sentdata> </sentdata>
    <sentcustomerid />
    <received></received>
    <sentdatetime></sentdatetime>
  </data>
  <call_result>
    <result> </result>
    <error />
  </call_result>
</api_result>
    
```

XML tag	Description
replyid	The max unique REPLYID for each incoming message
eventid	When sending an eventid is generated and returned in the result xml. This eventid can be for 1 or "infinity" message entries.
numfrom	The number of the user that sent the incoming message
receiveddata	The data that the user returned
received	The date and time the message was received
sentid	The unique sentid for the outgoing message
sentdata	The data sent to the user
sentdatetime	The date and time the message was sent
sentcustomerid	The customerid for the message when the data was sent
optout	Contains the value 1 if the receiveddata matched an opt out phrase or 0 if not

# GETTING YOUR GROUPS

There are 4 methods to retrieve your groups:

- Groups\_List\_DS\_DS (Accepts Dataset, returns Dataset)
- Groups\_List\_STR\_DS (Accepts XML string, returns Dataset)
- Groups\_List\_STR\_STR (Accepts XML string, returns XML string)
- Groups\_List\_DS\_STR (Accepts Dataset, returns XML string)

Each function accepts 3 parameters

- Username (String)
- Password (String)
- XML String or Dataset (depending on the function you choose)

The XML object looks like:

```
<options>
  <settings>
    <cols_returned> </cols_returned>
    <date_format> </date_format>
  </settings>
</options>
```

XML Tag	Description	Default
*Cols_Returned	The xml tags returned in the result. Options are: groupname, groupdesc, groupcreated. Each entry must be seperated with a comma.	Groupid
Date Format	The format of any dates that are returned.	dd/MMM/yyyy HH:mm:ss

(All entries marked with \* are Mandatory)

The output returned will be a dataset or XML in the following format:

```
<api_result>
  <data>
    <replyid></replyid>
    <groupid>118093</groupid>
    <groupname>the name of your group</groupname>
    <groupdesc>the desc of your group</groupdesc>
    <groupcreated>13/Apr/2009 13:00:05</groupcreated>
  </data>
  <call_result>
    <result> </result>
    <error />
  </call_result>
</api_result>
```

XML tag	Description
groupid	A unique GROUPID for each group in the system
Groupname	The name of your group
Groupdesc	The description you gave your group
groupcreated	When the group was created

# RETRIEVING - YOUR SHORT CODE COUNTS

There are 4 methods to retrieve the number of messages sent to your short codes.

- ShortCodeCount\_DS\_DS (Accepts Dataset, returns Dataset)
- ShortCodeCount\_STR\_DS (Accepts XML string, returns Dataset)
- ShortCodeCount\_STR\_STR (Accepts XML string, returns XML string)
- ShortCodeCount\_DS\_STR (Accepts Dataset, returns XML string)

Each function accepts 3 parameters

- Username (String)
- Password (String)
- XML String or Dataset (depending on the function you choose)

The XML object looks like

```
<options>
  <settings>
    <start_date>01/Oct/2011</start_date>
    <end_date>10/Oct/2011</end_date>
  </settings>
  <shortcodes>
    <shortcode>XXXXXX</shortcode>
  </shortcodes>
  <shortcodes>
    <shortcode>YYYYYY</shortcode>
  </shortcodes>
</options>
```

XML Tag	Description	Default
*start_date	The start date of the query period. Start_date can be set up to 12 months in the past.	-
*end_date	The end date of the query period. The start_date and end_date must be between 1 and 31 days.	-
shortcodes	The collection of short codes. If omitted, will return all short codes assigned to you with totals for the period.	-
shortcode	A short code number.	-

(All entries marked with \* are Mandatory)

The output returned will be a dataset or XML in the following format

```
<api_result>
  <data>
    <shortcode>XXXXX</shortcode>
    <count>200</count>
  </data>
  <data>
    <shortcode>YYYYY</shortcode>
    <count>45</count>
  </data>
  <call_result>
    <result>True</result>
    <error />
  </call_result>
</api_result>
```

XML tag	Description
shortcode	Your short code you received messages on.
count	The total count associated to the short code for the period.