



SOAP and HTTP GET API integration guide

The Messaging Centre
<http://www.tmcsm.com>
Integration guide version: 1.8
Gateway version: 1.0
September 2009

Contents

Contents	2
Introduction.....	3
Overview.....	3
Getting started.....	4
SOAP API.....	4
HTTP GET API.....	4
SMTP (Email) API.....	5
Important Security Notice.....	5
API Services	6
Send Service	6
Check Login.....	6
Check Credit	8
Hash Password.....	10
Send Message	11
Send Message 2	14
Send WAP Push.....	14
Send WAP Push 2.....	17
Get Message Job Codes	17
Get Message Job Codes 2	19
Get Message Status	19
Inbox Service	21
Get New Messages	21
Get New Messages Count.....	23
Get Message Text	24
Get Messages Digits	25
Reminder Service.....	27
Add Text	27
Add Wap Push.....	29
Other Reminder methods	32
Inbound number to Web page	32
Server IP	33
Other details	33
Sample code	33
Contact details	33
Other message sending options, tools and details	33
Troubleshooting	34
Glossary.....	35

Introduction

The Messaging Centre (TMC) SMS service provides a fully scalable, web-based SMS platform for communication with customers, suppliers, prospects, remote personnel or anyone with a compatible GSM device.

As well as our web based application site <http://www.tmcsm.com>, TMC SMS support a variety of options for integrating our SMS gateway with a third party application or system via the TMC SMS API.

This document is intended for developers and other technical staff and expects a basic understand of Information Technologies.

Overview

This document is intended for those users and their appointed technical staff to develop applications and integrate other systems that integrate with our SMS API for the sending, receiving and checking of SMS Messages and other related options.

It is recommended that you have signed up for an account at <http://www.tmcsm.com> as this will be necessary to test the solutions you prepare.

All messages are charged at your standard text bundle rate. There are no setup costs, monthly fees or other charges. See <http://www.tmcsm.com> for more details.

There are a number of ways to access our API gateway:

1. SOAP (XML) API
2. HTTP GET API
3. SMTP (Email) API

Details of how to use each of these API access methods along with a summary of each access method with links to more information can be found below.

Using the API it is possible to:

1. Send SMS messages (either single or in batches)
2. Send a WAP Push
3. Get a list of Message Job codes for SMS messages.
4. Get the status of a message
5. Get a list of new messages
6. Count the number of new messages
7. Get the content of a text message
8. Schedule a message for future delivery.

In addition our API can be accessed using the options listed in our “Other message sending options and tools” chapter below.

Examples and sample code can be found in the document below and by following the relevant links where indicated.

Please note that for all methods, both HTTP GET and SOAP, ALL parameters are required and must be passed or an error will occur.

Getting started

In order to use the TMC API it is necessary to have a TMC SMS account. Some technical knowledge and skills will also be required to implement solutions and functionality described below although the examples provided will help with this.

To register for an account please go to <http://www.tmcsm.com> and click on either the Free trial or Signup links to be taken to the corresponding signup page.

Once you have signed up you may wish to download some of the code samples (see the sample code chapter below) and read through the examples and method descriptions below.

It is not necessary to activate the API gateway – once you have a TMC SMS account the API can be used immediately.

SOAP API

The SOAP API allows for interaction with our API using the XML Based Simple Object Access Protocol (SOAP). Requests are sent via the HTTP (or HTTPS) POST method to our gateway and results are returned in the HTTP response.

For security all requests can be sent using the HTTPS protocol so that the message body is encrypted against any form of interception to protect your message and password hash.

Please see the API Services chapter below for more details. Online documentation for the SOAP Web Services API (implemented using Microsoft .NET technology) can also be found at the following locations:

1. For sending messages and other sending options please go to <https://www.tmcsm.com/soap/SendService.aspx>
2. To check for new messages and other message retrieval options please go to <https://www.tmcsm.com/soap/InboxService.aspx>
3. For adding a scheduled message send please go to <https://www.tmcsm.com/soap/ReminderService.aspx>

Note: Please note that it is not possible to list or cancel a schedule using the API – to do this you must log into the main application site – <http://www.tmcsm.com>.

HTTP GET API

The HTTP GET API allows for interaction with our API using the HTTP GET method by passing a standard HTTP URL to our gateway. Requests are sent via the

HTTP GET method to our gateway and results are returned in the HTTP response from the server.

The HTTP GET API supports all of the functionality of the SOAP API above however parameters are sent in the URL query string to the server rather than by in a HTTP POST body as the SOAP API uses to send requests. Please see the API Services chapter below for more details on how to use the HTTP GET service.

Please refer to the important security notice below for more details on sending requests without HTTPS.

Note: Please note that all variable values for the query string must be URL encoded as some HTTP functions do not automatically do this. Only the variable values (and not the whole URL) should be URL encoded. Most languages have a URL encoding function, check online or contact support for more details.

Also please note that certain URL encoded values must be in UTF-8 format or they will not appear in the end user message. This only affects certain characters such as symbol characters (£,\$ etc), if you are experiencing missing characters then it is most likely you are using the wrong encoding format.

SMTP (Email) API

Unlike the SOAP API and HTTP GET API methods of accessing our gateway the SMTP (Email) method of accessing our gateway is not strictly an API method and can be used with standard email clients (e.g. Outlook, Thunderbird, Eudora).

Some third party applications and systems do support messaging sending via SMTP and not via other methods above and so our gateway can be used to integrate our gateway with these systems.

For more details on how to access the SMTP API please refer to the Email to SMS documentation http://www.tmcsm.co.uk/docs/tmc_email_to_sms_guide.pdf.

Note: The services described in the API Services chapter below **DO NOT** apply to the SMTP API. For details on how to use the SMTP API please refer to the above document.

Important Security Notice

Please note that while it is possible to use the SMTP, HTTP and HTTPS protocols with our server it is recommended that you use HTTPS where possible as this will automatically encrypt the request body that contains the message, a hash of the account password and other details.

It is only possible to use HTTPS effectively with the GET and SOAP methods of accessing our service and so this is the recommended option of using the API where possible. We have supplied the other less secure methods to support users that are

happy to accept the relatively minimal security risk of not using HTTPS on their messages.

Note: When passing parameters in a GET query string using the HTTPS protocol HTTPS should detach the query string from the URL end encrypt it before relaying it to the destination server where it will be decrypted at the other end (source MSDN).

Note: Although the account password is hashed it is NOT a one time only hash and so can be reused with each request to send messages and because of this the HTTP SOAP, HTTP Get and Email methods of using the service are less secure then the HTTPS SOAP and HTTPS GET methods.

API Services

The TMC API supports 3 services for accessing the gateway to send, receive and schedule SMS messages. All 3 services can be used with both the HTTP SOAP and HTTP GET methods (for SMTP/Email messages please see the above SMTP (Email) API Chapter).

Please see read the service descriptions below for syntax, parameters, examples and links to sample code, comments and other details.

Send Service

The send service allows for the sending of SMS messages outbound from the TMC SMS gateway to end user mobiles and other devices. The service provides the following methods for interacting with the service.

Service locations:

SOAP requests should be sent via HTTP POST to:

<https://www.tmcsmms.co.uk/soap/SendService.asmx>

HTTP GET requests should be sent to:

<https://www.tmcsmms.co.uk/api/SendService.aspx>

Notes:

The Confirmation parameter (see parameter details below) can accept a value in either hex (e.g. 0x4) or decimal (e.g. 4). It is recommended you use decimal to avoid any issues.

Check Login

The CheckLogin method can be used to validate a username and password and check the status of the users account.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsmms.co.uk
pwd	An MD5 hash in Base 64 format of	hjdsfhuewhf==

	the password defined or supplied by the TMC. This hash should be generated using the HashPassword method below.	
--	---	--

Return value(s)

This method returns one of the following values

Name	Description
Success	The users login details are valid and these details can be used to login to the TMC SMS Gateway (note CheckCredit should be used to check that the user has credit)
InternalError	An internal system error
UserNotFound	No matching user entry has been found for the email address supplied in the email parameter.
UserNotActive	This user is not currently active within the TMC SMS system.
CompanyTermsAndConditions	Company did not agree to the terms and conditions (this is done by logging into the application site http://www.tmcsmcs.co.uk)
UserTermsAndConditions	User did not agree to the terms and conditions (this is done by logging into the application site http://www.tmcsmcs.co.uk)
SetupGuide	

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsmcs.co.uk
Content-Type: text/xml; charset=utf-8
Content-Length: 540
SOAPAction: "co.uk.tmcsmcs.soap/CheckLogin"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <CheckLogin xmlns="co.uk.tmcsmcs.soap">
      <email>test@tmcsmcs.co.uk</email>
      <pwd>hjdsfhuewhf==</pwd>
    </CheckLogin>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 430

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <CheckLoginResponse xmlns="co.uk.tmcSMS.soap">
      <CheckLoginResult>Success</CheckLoginResult>
    </CheckLoginResponse>
  </soap:Body>
</soap:Envelope>

```

For more details and examples of this SOAP API function please go to:
<https://www.tmcSMS.co.uk/soap/SendService.asmx?op=CheckLogin>

HTTP GET Method

<https://www.tmcSMS.co.uk/api/SendService.aspx?method=CheckLogin&email=<email>&pwd=<pwd>>

For example

<https://www.tmcSMS.co.uk/api/SendService.aspx?method=CheckLogin&email=test@tmcSMS.co.uk&pwd=hjdsfhuewhf==>

Check Credit

The CheckCredit method can be used to check whether the user has enough credit to send a message without actually sending the message. It can also be used to validate other parameters such as recipients.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcSMS.co.uk
Pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
recipients	A List of mobile numbers separated with a semi-colon (;) character. New line characters are not allowed in this field. NOTE: This field is required to verify there is enough credit for	+44797912346 or 07997912346

	the intended recipients.	
body	Message to be sent in either Unicode or ASCII format	This is a test message
unicode	Indicates that message is encoded using the Unicode character set. Value is either true or false.	true (if msg is Unicode) or false (otherwise)

Return value(s)

This method returns one of the following values

Name	Description
Success	The users login details are valid and these details can be used to login to the TMC SMS Gateway (note CheckCredit should be used to check that the user has credit)
InternalError	An internal system error
InvalidRecipients	The recipients list contains one or more invalid recipients.
CompanyNoCredit	The Company that this user is a member of does not have enough credit to send the message to the above recipients.
UserNoCredit	The User does not have enough credit allocated to send the message to the above recipients.
CompanyExpired	The Company does not have any more credit for message sending.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmc sms.co.uk
Content-Type: text/xml; charset=utf-8
Content-Length: 620
SOAPAction: "co.uk.tmc sms.soap/CheckCredit"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <CheckCredit xmlns="co.uk.tmc sms.soap">
      <email>test@tmc sms.com</email>
      <pwd>hjdsfhuewhf==</pwd>
      <recipients>07958123456;+447958456879</recipients>
      <body>This is a test</body>
      <unicode>>false</unicode>
    </CheckCredit>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 406

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <CheckCreditResponse xmlns="co.uk.tmcSMS.soap">
      <CheckCreditResult>Success</CheckCreditResult>
    </CheckCreditResponse>
  </soap:Body>
</soap:Envelope>

```

For more details and examples of this SOAP API function please go to:
<https://www.tmcSMS.co.uk/soap/SendService.asmx?op=CheckCredit>

HTTP GET Option

<https://www.tmcSMS.co.uk/api/SendService.aspx?method=CheckCredit&email=<email>&pwd=<pwd>&recipients=<recipients>&body=<body>&unicode=<unicode>>

For example

<https://www.tmcSMS.co.uk/api/SendService.aspx?method=CheckCredit&email=test@tmcSMS.co.uk&pwd=hjdsfhuewhf==&recipients=07958123456;+447958456879&body=This%20is%20a%20Test&unicode=false>

Hash Password

This method generates an MD5 Base 64 hash of the password supplied via the pwd parameter. The password should be a valid password for the TMC SMS system.

Parameters

Name	Description	Example
Pwd	The actual valid TMC password for the user to be used with the password parameter of all the other API methods defined in this document.	validtmcuserpassword

Return value(s)

A string consisting of the MD5 hashed password.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```

POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsmcs.co.uk
Content-Type: text/xml; charset=utf-8
Content-Length: 520
SOAPAction: "co.uk.tmcsmcs.soap/HashPassword"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <HashPassword xmlns="co.uk.tmcsmcs.soap">
      <pwd>validtmcuserpassword</pwd>
    </HashPassword>
  </soap:Body>
</soap:Envelope>

```

A typical response to the above request would be:

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 380

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <HashPasswordResponse xmlns="co.uk.tmcsmcs.soap">
      <HashPasswordResult>hjdsfhuewhf==</HashPasswordResult>
    </HashPasswordResponse>
  </soap:Body>
</soap:Envelope>

```

For more details and examples of this SOAP API function please go to:
<https://www.tmcsmcs.co.uk/soap/SendService.asmx?op=HashPassword>

HTTP GET Option

<https://www.tmcsmcs.co.uk/api/SendService.aspx?method=HashPassword&pwd=<pwd>>

For example

<https://www.tmcsmcs.co.uk/api/SendService.aspx?method=HashPassword&pwd=validtmcuserpassword>

Send Message

When sending messages through the TMC gateway using the SOAP API there are 2 methods of sending out SMS Messages. Both methods are identical with the exception of the return value where:

- 1) With SOAP SendMessage returns a series of SOAP string elements in its response with a string element corresponding to each message job code.
- 2) With SOAP SendMessage2 returns a single SOAP string element in its response that contains a comma separated list of message job codes.

- 3) For HTTP GET there is only one method – SendMessage – which returns a comma separated list of message job codes (note there is no HTTP GET SendMessage2 method and the result is the same format as the SOAP SendMessage2 method).

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsm.co.uk
Pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
recipients	A List of mobile numbers separated with a semi-colon (;) character. New line characters are not allowed in this field. NOTE: This field is required to verify there is enough credit for the intended recipients.	+44797912346 or 07997912346
body	Message to be sent in either Unicode or ASCII format	This is a test message
reference	A reference that you can later use to identify this message internally	Website alert
source	The “from” name of this message (like an email from) – can be one of the following literal values Mobile, CompanyName, NickName, Signature or InboundNumber and the value is then taken from the users profile within the system.	NickName
confirm	Confirmation of message delivery, Binary operation of values 0 for nothing, 2 for Mobile, 4 for Email	4
unicode	Indicates that message is encoded using the Unicode character set. Value is either true or false.	true (if msg is Unicode) or false (otherwise)

Return value(s)

This method can return one of 3 forms of value:

- 1) If there is an error then null is returned (blank in HTTP GET)
- 2) If there are less than 20 recipients in the request then a list of Message job codes is returned – the result list is either a comma separated list or a series of SOAP string elements (see comments above and below about the difference between SendMessage and SendMessage2)
- 3) If there are more than 20 recipients in the combined SOAP request then a single batch message identifier is returned that can be used with the GetMessageJobCodes method to return a full list of message job

codes. Please note that in this scenario messages are sent in the background and all message job codes may not be immediately available.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 650
SOAPAction: "co.uk.tmcsm.com/SendMessage"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SendMessage xmlns="co.uk.tmcsm.com">
      <email>test@tmcsm.com</email>
      <pwd>hjdsfhuewhf==</pwd>
      <recipients>+44797912346 or 07997912346</recipients>
      <body>This is a test message</body>
      <reference>Website alert</reference>
      <source>NickName </source>
      <confirm>4</confirm>
      <unicode>>true</unicode>
    </SendMessage>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 450

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SendMessageResponse xmlns="co.uk.tmcsm.com">
      <SendMessageResult>
        <string>9937BE8D-EDAD-4736-97B2-7824AA5234DD</string>
        <string>8837AE8D-EFAD-4666-97B2-7824AA5564DD</string>
      </SendMessageResult>
    </SendMessageResponse>
  </soap:Body>
</soap:Envelope>
```

For more details and examples of this SOAP API function please go to:
<https://www.tmcsm.com/soap/SendService.asmx?op=SendMessage>

HTTP GET Option

<https://www.tmcsmms.co.uk/api/SendService.aspx?method=SendMessage&email=<email>&pwd=<pwd>&recipients=<recipients>&body=<body>&reference=<reference>&source=<source>&confirm=<confirm>&unicode=<unicode>>

For example

<https://www.tmcsmms.co.uk/api/SendService.aspx?method=SendMessage&email=test@tmcsmms.co.uk&pwd=hjdsfhuewhf==&recipients=+447958123456;0014567899;07988123444&body=This%20Is%20A%20Test&reference=Message1&source=Mobile&confirm=2&unicode=False>

Send Message 2

The SendMessage2 method only applies to the SOAP API gateway and returns a comma separated list of Message job codes as a single soap response rather than the series of soap string elements returned by the SOAP version of SendMessage.

An SendMessage 2 example can be found here:

<https://www.tmcsmms.co.uk/soap/SendService.asmx?op=SendMessage2>

Note: Send Message 2 does not apply to HTTP GET where returning a csv list (the only difference between SendMessage and SendMessage2) is the only result format.

Send WAP Push

When sending wap push messages through the TMC gateway using the SOAP API there are 2 methods of sending out messages. Both methods are identical with the exception of the return value where:

- 1) SOAP SendWapPush returns a series of SOAP string elements in its response with a string element corresponding to each message job code.
- 2) SOAP SendWapPush2 returns a single SOAP string element in its response that contains a comma separated list of message job codes.
- 3) For HTTP GET there is only one method – SendWapPush – which returns a comma separated list of message job codes (note there is no HTTP GET SendWapPush2 method and the result is the same format as the SOAP SendWapPush2 method).

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsmms.co.uk
Pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
recipients	A List of mobile numbers separated with a semi-colon (;) character. New line characters are not allowed in this field.	+44797912346 or 07997912346

	NOTE: This field is required to verify there is enough credit for the intended recipients.	
body	Message to be sent in either Unicode or ASCII format	This is a test message
url	The URL to be sent with the wap push (see glossary)	http://www.tmcsm.com
reference	A reference that you can later use to identify this message internally	Website alert
source	The "from" name of this message (like an email from) – can be one of the following literal values Mobile, CompanyName, NickName, Signature or InboundNumber and the value is then taken from the users profile within the system.	NickName
confirm	Confirmation of message delivery, Binary operation of values 0 for nothing, 2 for Mobile, 4 for Email	4
unicode	Indicates that message is encoded using the Unicode character set. Value is either true or false.	true (if msg is Unicode) or false (otherwise)

Return value(s)

This method can return one of 3 forms of value:

- 1) If there is an error then null is returned (blank in HTTP GET)
- 2) If there are less than 20 recipients in the request then a list of Message job codes is returned – the result list is either a comma separated list or a series of SOAP string elements (see comments above and below about the difference between SendWapPush and SendWapPush2)
- 3) If there are more than 20 recipients in the combined SOAP request then a single message identifier is returned that can be used with the GetMessageJobCodes method to return a full list of message job codes. Please note that in this scenario, messages are sent in the background and all message job codes may not be immediately available.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 580
SOAPAction: "co.uk.tmcsm.com/soap/SendWapPush"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

```

<soap:Body>
  <SendWapPush xmlns="co.uk.tmcSMS.soap">
    <email>test@tmcSMS.com</email>
    <pwd>hjdsfhuewhf==</pwd>
    <recipients>+44797912346 or 07997912346</recipients>
    <body>This is a test</body>
    <url>http://www.tmcSMS.com</url>
    <reference>Website alert</reference>
    <source> NickName </source>
    <confirm>0x4</confirm>
    <unicode>>true</unicode>
  </SendWapPush>
</soap:Body>
</soap:Envelope>

```

A typical response to the above request would be:

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 430

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SendMessageResponse xmlns="co.uk.tmcSMS.soap">
      <SendMessageResult>
        <string>9937BE8D-EDAD-4736-97B2-7824AA5234DD</string>
        <string>8837AE8D-EFAD-4666-97B2-7824AA5564DD</string>
      </SendMessageResult>
    </SendMessageResponse>
  </soap:Body>
</soap:Envelope>

```

For more details and examples of this SOAP API function please go to:
<https://www.tmcSMS.co.uk/soap/Service.asmx?op=SendWapPush>

HTTP GET Option

<https://www.tmcSMS.co.uk/api/Service.aspx?method=SendMessage&email=<email>&pwd=<pwd>&recipients=<recipients>&body=<body>&url=<url>&reference=<reference>&source=<source>&confirm=<confirm>&unicode=<unicode>>

For example

<https://www.tmcSMS.co.uk/api/Service.aspx?method=SendMessage&email=test@tmcSMS.co.uk&pwd=hjdsfhuewhf==&recipients=+447958123456;0014567899;07988123444&body=This%20Is%20A%20Test&url=http://www.tmcSMS.co.uk&reference=Message%201&source=Mobile&confirm=2&unicode=False>

Send WAP Push 2

The SendWAPPush2 method only applies to the SOAP API gateway and returns a comma separated list of Message job codes as a single soap response rather than the series of soap string elements returned by the SOAP version of SendWAPPush.

A SendWAPPush2 example can be found here:

<https://www.tmcsm.co.uk/soap/Service.asmx?op=SendWapPush2>

Note: SendWAPPush2 does not apply to HTTP GET where returning a csv list (which is the only difference between SendWAPPush and SendWAPPush2) is the only behaviour.

Get Message Job Codes

This method returns a list of job codes for a SendMessage, SendMessage2, SendWapPush or SendWapPush2 request. This is particularly necessary when a request has more than 20 recipients (see above for more details).

Note: this method requires the batch message identifier returned from a batch job of greater than 20 recipients – it does not accept standard message identifiers from smaller jobs.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsm.co.uk
pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
jobcode	Batch Message identifier returned by SendMessage, SendMessage2, SendWapPush or SendWapPush2	9937BE8D-EDAD-4736-97B2-7824AA5234DD

Return value(s)

For SOAP requests, a SOAP result list of string elements, with each element containing a message identifier for an SMS message.

For HTTP GET a comma separated list of message identifiers is returned.

Note: There is no comma separated SOAP result option at this time. If you need this functionality please ensure you send messages in batches of less than 20 and the message identifiers are then returned in the response from the relevant method as defined above.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 420
SOAPAction: "co.uk.tmcsm.com/GetMessageJobCodes"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageJobCodes xmlns="co.uk.tmcsm.com">
      <email>test@tmcsm.com</email>
      <pwd>hjdsfhuewhf==</pwd>
      <jobcode>9937BE8D-EDAD-4736-97B2-7824AA5234DD</jobcode>
    </GetMessageJobCodes>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 380

<?xml version="1.0" encoding="utf-8"?>
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 436

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageJobCodesResponse xmlns="co.uk.tmcsm.com">
      <GetMessageJobCodesResult>
        <string>9937BE8D-EDAD-4736-97B2-7824AA5234DD</string>
        <string>9897BE8D-EJAD-4736-97B2-4924AA6634JD</string>
      </GetMessageJobCodesResult>
    </GetMessageJobCodesResponse>
  </soap:Body>
</soap:Envelope>
```

For more details and examples of this SOAP API function please go to:
<https://www.tmcsm.com/soap/SendService.asmx?op=GetMessageJobCodes>

HTTP GET Option

<https://www.tmcsm.com/api/SendService.aspx?method=GetMessageJobCodes&email=<email>&pwd=<pwd>&jobcode=<jobcode>>

For example

<https://www.tmcsm.com/api/SendService.aspx?method=GetMessageJobCodes&email=test@tmcsm.com&pwd=hjdsfhuewhf==&jobcode=9937BE8D-EDAD-4736-97B2-7824AA5234DD>

Get Message Job Codes 2

The GetMessageJobCodes2 method only applies to the SOAP API gateway and returns a comma separated list of Message job codes as a single soap response rather than the series of soap string elements returned by the SOAP version of GetMessageJobCodes.

A GetMessageJobCodes2 example can be found here:

<https://www.tmcsmc.co.uk/soap/SendService.asmx?op=GetMessageJobCodes2>

Note: GetMessageJobCodes2 does not apply to HTTP GET where returning a csv list (which is the only difference between GetMessageJobCodes and GetMessageJobCodes2) is the only behaviour.

Get Message Status

This method returns the status of the a SMS message sent via SendMessage, SendMessage2, SendWapPush or SendWapPush2.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsmc.co.uk
pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
jobcode	Message identifier returned by SendMessage, SendMessage2, SendWapPush or SendWapPush2 or retrieved via the GetMessageCodes function.	9937BE8D-EDAD-4736-97B2-7824AA5234DD

Return value(s)

This method returns one of the following values

Name	Description
Sent	The users login details are valid and these details can be used to login to the TMC SMS Gateway (note CheckCredit should be used to check that the user has credit)
Delivered	An internal system error
Undelivered	The recipients list contains one or more invalid recipients.
Queued	The Company that this user is a member of does not have enough credit to send the message to the above recipients.

Failed	The User does not have enough credit allocated to send the message to the above recipients.
Expired	The Company does not have any more credit for message sending.
InternalError	An internal system error has occurred
UserNotFound	The user identified by the email and password parameters has not been found in the system.
UserNotActive	The user identified by the email address and password is not active.
CompanyTermsAndConditions	Company did not agree to terms and conditions.
UserTermsAndConditions	User did not agree to the terms and conditions.
MessageNotFound	The message identified by the job code parameter was not found.
InvalidCompany	Invalid company.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 429
SOAPAction: "co.uk.tmcsm.com/GetMessageStatus"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageStatus xmlns="co.uk.tmcsm.com">
      <email>test@tmcsm.com</email>
      <pwd>hjdsfhuewhf==</pwd>
      <jobcode>9937BE8D-EDAD-4736-97B2-7824AA5234DD</jobcode>
    </GetMessageStatus>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 420

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageStatusResponse xmlns="co.uk.tmcsm.com">
      <GetMessageStatusResult>Delivered</GetMessageStatusResult>
```

```
</GetMessageStatusResponse>
</soap:Body>
</soap:Envelope>
```

For more details and examples of this SOAP API function please go to:
<https://www.tmcsm.com/soap/SendService.aspx?op=GetMessageStatus>

HTTP GET Option

<https://www.tmcsm.com/api/SendService.aspx?method=GetMessageStatus&email=<email>&pwd=<pwd>&jobcode=<jobcode>>

For example

<https://www.tmcsm.com/api/SendService.aspx?method=GetMessageStatus&email=test@tmcsm.com&pwd=hjdsfhuewhf==&jobcode=9937BE8D-EDAD-4736-97B2-7824AA5234DD>

Inbox Service

The inbox service allows for the receiving of inbound SMS messages through the TMC SMS gateway to that have been sent by user mobiles and other devices. The service provides the following methods for interacting with the service.

Service locations:

SOAP requests should be sent via HTTP POST to:

<https://www.tmcsm.com/soap/InboxService.aspx>

HTTP GET requests should be sent to:

<https://www.tmcsm.com/api/InboxService.aspx>

Get New Messages

GetNewMessages returns a list of new message identifiers that can be used with the GetMessageText function to retrieve the actual message content.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsm.com
pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==

Return value(s)

For SOAP requests, a SOAP result list of string elements, with each element containing a message identifier for an SMS message.

For HTTP GET a comma separated list of message identifiers is returned.

Note: There is no comma separated SOAP result option at this time. If you need this functionality please contact support.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 420
SOAPAction: "co.uk.tmcsm.com/GetNewMessages"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetNewMessages xmlns="co.uk.tmcsm.com">
      <email>test@tmcsm.com</email>
      <pwd>hjdsfhuewhf==</pwd>
    </GetNewMessages>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 400

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetNewMessagesResponse xmlns="co.uk.tmcsm.com">
      <GetNewMessagesResult>
        <int>123456</int>
        <int>78978676</int>
      </GetNewMessagesResult>
    </GetNewMessagesResponse>
  </soap:Body>
</soap:Envelope>
```

Please note that although the above example specifies integer (int) values as the result list these may be string values.

For more details and examples of this SOAP API function please go to:
<https://www.tmcsm.com/soap/InboxService.asmx?op=GetNewMessages>

HTTP GET Option

[https://www.tmcsm.com/api/InboxService.aspx?method=GetNewMessages
&email=<email>&pwd=<pwd>](https://www.tmcsm.com/api/InboxService.aspx?method=GetNewMessages&email=<email>&pwd=<pwd>)

For example

<https://www.tmcsm.com/api/InboxService.aspx?method=GetNewMessages&email=test@tmcsm.com&pwd=hjdsfhuewhf==>

Get New Messages Count

GetNewMessagesCount returns the number of new messages waiting to be read. The GetNewMessages and GetMessageText functions can then be used to retrieve the actual messages and their content.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsm.com
pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==

Return value(s)

An integer value that is either the number of new messages found or -1 indicating an error.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 420
SOAPAction: "co.uk.tmcsm.com/soap/GetNewMessagesCount"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetNewMessagesCount xmlns="co.uk.tmcsm.com">
      <email>test@tmcsm.com</email>
      <pwd>hjdsfhuewhf==</pwd>
    </GetNewMessagesCount>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 400

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetNewMessagesCountResponse xmlns="co.uk.tmcSMS.soap">
      <GetNewMessagesCountResult>2</GetNewMessagesCountResult>
    </GetNewMessagesCountResponse>
  </soap:Body>
</soap:Envelope>
```

For more details and examples of this SOAP API function please go to:
<https://www.tmcSMS.co.uk/soap/InboxService.asmx?op=GetNewMessagesCount>

HTTP GET Option

<http://www.tmcSMS.co.uk/api/InboxService.aspx?method=GetNewMessagesCount&email=<email>&pwd=<pwd>>

For example

<https://www.tmcSMS.co.uk/api/InboxService.aspx?method=GetNewMessagesCount&email=test@tmcSMS.com&pwd=hjdsfhuewhf==>

Get Message Text

GetMessageText returns the text message that matches the message identifier supplied via the id parameter.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcSMS.co.uk
pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
id	Message identifier	12345

Return value(s)

The message content or null if there is an error.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcSMS.co.uk
Content-Type: text/xml; charset=utf-8
Content-Length: 420
SOAPAction: "co.uk.tmcSMS.soap/GetNewMessageText"
```



```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageText xmlns="co.uk.tmcSMS.soap">
      <email>test@tmcSMS.com</email>
      <pwd>hjdsfhuewhf==</pwd>
      <id>12345</id>
    </GetMessageText>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 410

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageTextResponse xmlns="co.uk.tmcSMS.soap">
      <GetMessageTextResult>This is a
test</GetMessageTextResult>
    </GetMessageTextResponse>
  </soap:Body>
</soap:Envelope>
```

For more details and examples of this SOAP API function please go to:
<https://www.tmcSMS.co.uk/soap/InboxService.asmx?op=GetMessageText>

HTTP GET Option

<http://www.tmcSMS.co.uk/api/InboxService.aspx?method=GetMessageText&mail=<email>&pwd=<pwd>&id=<id>>

For example

<https://www.tmcSMS.co.uk/api/InboxService.aspx?method=GetMessageText&email=test@tmcSMS.com&pwd=hjdsfhuewhf==&id=12345>

Get Messages Digits

GetNewMessagesDigits function returns the sender name/number of the message that matches the message identifier supplied via the id parameter. This can be a mobile number, name or other value.

Parameters

Name	Description	Example
------	-------------	---------

email	A valid email address that has been designated as a user login	test@tmcsms.co.uk
pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
id	Message identifier	112233

Return value(s)

The message senders name/number or null if there is an error.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/SendService.asmx HTTP/1.1
Host: www.tmcsms.co.uk
Content-Type: text/xml; charset=utf-8
Content-Length: 440
SOAPAction: "co.uk.tmcsms.soap/GetNewMessageDigits"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageDigitsResponse xmlns="co.uk.tmcsms.soap">
      <email>test@tmcsms.com</email>
      <pwd>hjdsfhuewhf==</pwd>
      <id>112233</id>
    </GetMessageDigitsResponse>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 375

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMessageDigitsResponse xmlns="co.uk.tmcsms.soap">
      <GetMessageDigitsResult>+447958456789</GetMessageDigitsResult>
    </GetMessageDigitsResponse>
  </soap:Body>
</soap:Envelope>
```

For more details and examples of this SOAP API function please go to:
<https://www.tmcsms.co.uk/soap/InboxService.asmx?op=GetMessagesDigits>

HTTP GET Option

http://www.tmcsm.com/api/InboxService.aspx?method=GetMessageDigits
&email=<email>&pwd=<pwd>&id=<id>

For example

https://www.tmcsm.com/api/InboxService.aspx?method=GetMessageDigits
&email=test@tmcsm.com&pwd=hjdsfhuewhf==&id=12233

Reminder Service

The reminder service allows for the scheduled sending of repeat SMS messages outbound from the TMC SMS gateway to end user mobiles and other devices at defined intervals. The service provides the following methods for interacting with the service.

Service locations:

SOAP requests should be sent via HTTP POST to:

<https://www.tmcsm.com/soap/ReminderService.asmx>

HTTP GET requests should be sent to:

<https://www.tmcsm.com/api/ReminderService.aspx>

Add Text

The AddText method is used to set a SMS Text message at a specified frequency.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcsm.com
pwd	An MD5 hash in Base 64 format of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	hjdsfhuewhf==
next	Date/time to send the next message	26-june-2007 13:10
frequency	Interval between message sending, one of: Once – Only one message sent Daily – message sent daily Weekly – message sent at weekly intervals on the same day as next Weekdays – message sent every weekday Monday to Friday. Monthly – message sent once a month on the day specified by next	Daily

	<p>Annually – message sent once a year on the day and month specified by next</p> <p>Note: Message sending will start at the date/time specified by the next parameter</p>	
recipients	<p>A List of mobile numbers separated with a semi-colon (;) character. New line characters are not allowed in this field.</p> <p>NOTE: This field is required to verify there is enough credit for the intended recipients.</p>	+44797912346 or 07997912346
body	Message to be sent	This is a test
source	<p>The “from” name of this message (like an email from) – can be one of the following literal values Mobile, CompanyName, NickName, Signature or InboundNumber and the value is then taken from the users profile within the system.</p>	NickName
confirm	<p>Confirmation of message delivery 0 for no confirmation, 2 for Mobile and 4 for Email</p>	4
unicode	Unicode encoding if value is true	true

Return value(s)

A single message identifier or 0 on error.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/ReminderService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 520
SOAPAction: "co.uk.tmcsm.com/soap/AddText"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddText xmlns="co.uk.tmcsm.com">
      <email>test@tmcsm.com</email>
      <pwd>hjdsfhuewhf==</pwd>
    </AddText>
  </soap:Body>
</soap:Envelope>
```

```

<next>26-June-2007 13:00</next>
<frequency>Daily</frequency>
<recipients>+447948123456;+447956899333</recipients>
<body>This is a test</body>
<source>Mobile</source>
<confirm>0x0</confirm>
<unicode>>false</unicode>
</AddText>
</soap:Body>
</soap:Envelope>

```

A typical response to the above request would be:

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 380

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddTextResponse xmlns="co.uk.tmcSMS.soap">
      <AddTextResult>9937BE8D-EDAD-4736-97B2-7824AA5234DD
    </AddTextResult>
    </AddTextResponse>
  </soap:Body>
</soap:Envelope>

```

For more details and examples of this SOAP API function please go to:
<https://www.tmcSMS.co.uk/soap/ReminderService.asmx?op=AddText>

HTTP GET Option

<https://www.tmcSMS.co.uk/api/ReminderService.aspx?method=AddText&email=<email>&pwd=<pwd>&next=<next>&frequency=<frequency>&recipient=<recipients>&body=<body>&source=<source>&confirm=<confirm>&unicode=<unicode>>

For example

<https://www.tmcSMS.co.uk/api/ReminderService.aspx?method=AddText&email=test@tmcSMS.com&pwd=hjdsfhuewhf==&next=26-June-2006%2013:00&frequency=Daily&recipient=+447958123456;+4478988999&body=This%20Is%20A%20Test&source=Mobile&confirm=0&unicode=false>

Add Wap Push

The AddText method is used to set a SMS Wap push message at a specified frequency.

Parameters

Name	Description	Example
email	A valid email address that has been designated as a user login	test@tmcSMS.co.uk
pwd	An MD5 hash in Base 64 format	hjdsfhuewhf==

	of the password defined or supplied by the TMC. This hash can be generated using the HashPassword method below.	
next	Date/time to send the next message	26-june-2007 13:10
frequency	Interval between message sending, one of: Once – Only one message sent Daily – message sent daily Weekly – message sent at weekly intervals on the same day as next Weekdays – message sent every weekday Monday to Friday. Monthly – message sent once a month on the day specified by next Annually – message sent once a year on the day and month specified by next Note: Message sending will start at the date/time specified by the next parameter	Daily
recipients	A List of mobile numbers separated with a semi-colon (;) character. New line characters are not allowed in this field. NOTE: This field is required to verify there is enough credit for the intended recipients.	+44797912346 or 07997912346
body	Message to be sent	This is a test
url	URL to be sent with the wap push	http://www.tmcsm.com
source	The “from” name of this message (like an email from) – can be one of the following literal values Mobile, CompanyName, NickName, Signature or InboundNumber and the value is then taken from the users profile within the system.	NickName
confirm	Confirmation of message delivery 0 for no confirmation, 2 for Mobile and 4 for Email	4
unicode	Unicode encoding if value is true	true

Return value(s)

A single message identifier or 0 on error.

SOAP 1.1 Example

The following example is a typical SOAP 1.1 example – see the URL below for more details, a SOAP 1.2 example and other details. The SOAP request should be posted to the above SOAP URL using HTTP POST.

```
POST /soap/ReminderService.asmx HTTP/1.1
Host: www.tmcsm.com
Content-Type: text/xml; charset=utf-8
Content-Length: 520
SOAPAction: "co.uk.tmcsm.com/AddWapPush"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddWapPush xmlns="co.uk.tmcsm.com">
      <email>test@tmcsm.com</email>
      <pwd>hjdsfhuewhf==</pwd>
      <next>26-June-2007 13:00</next>
      <frequency>Daily</frequency>
      <recipients>+447948123456;+447956899333</recipients>
      <body>This is a test</body>
      <url>http://www.tmcsm.com</url>
      <source>Mobile</source>
      <confirm>0x0</confirm>
      <unicode>>false</unicode>
    </AddWapPush>
  </soap:Body>
</soap:Envelope>
```

A typical response to the above request would be:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 420

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddTextResponse xmlns="co.uk.tmcsm.com">
      <AddTextResult>9937BE8D-EDAD-4736-97B2-7824AA5234DD
</AddTextResult>
    </AddTextResponse>
  </soap:Body>
</soap:Envelope>
```

For more details and examples of this SOAP API function please go to:
<https://www.tmcsm.com/soap/ReminderService.asmx?op=AddWapPush>

HTTP GET Option

<https://www.tmcsm.com/api/ReminderService.aspx?method=AddText&email=<email>&pwd=<pwd>&next=<next>&frequency=<frequency>&recipient=>

[<recipients>&body=<body>&source=<source>&confirm=<confirm>&unicode=<unicode>](#)

For example

<https://www.tmcsm.com/api/ReminderService.aspx?method=AddText&email=test@tmcsm.com&pwd=hjdsfhuewhf=&next=26-June-2006%2013:00&frequency=Daily&recipient=+447958123456;+4478988999&body=This%20Is%20A%20Test&source=Mobile&confirm=0&unicode=false>

Other Reminder methods

Please note that there are no methods to remove or list reminder services available at the moment. These activities must be performed via the application site <http://www.tmcsm.com> – please contact support if you require this functionality.

Inbound number to Web page

By default inbound messages are delivered to your TMC or Email mailbox based on the options set in your TMC control panel.

For developers looking to process inbound messages (e.g. to process an automated, customised response, store in another system or process the value) we can forward the message to a target web page URL where a script has been defined. e.g.

http://www.yoursite.com/inbound_message.php

The message is posted to the target URL using HTTP Post method. The following variables are posted to the web page :

Field	Value	Example
originator	The sender value, typically their mobile number	07958123456
body	The message that was received to the inbound number	This is a test
notificationType	A method identifier used to differentiate between Post calls. Currently this will always be “MessageReceived”	MessageReceived
recipient	The inbound number that the message was sent to.	0788123456

You can set the callback URL in the company details area of the control panel by going to Admin > Company > Details from the left hand menu and then entering the URL in the “Notify URL” box and click save.

Server IP

For security you may wish to verify the IP address of the inbound call to your Notification URL (See above Inbound number to Web page for details) or for other security purposes. You can also use this URL for unblocking firewall ports and other purposes as required.

The IP address of the TMC servers is currently 92.63.130.67. As with any Internet IP address this is subject to change without notice however it is a fixed IP and there are no plans to change it at this time.

If you are having an issue contact TMC or try pinging the domain name to discover the current IP address.

Other details

Please note that messages can be up-to 160 characters in length. This applies to all the methods described above where applicable. If a message is larger than 160 characters then it will be broken up into packets which are typically then concatenated at on the target handset or device. Each packet would be chargeable at standard TMC message rates.

Sample code

We have provided the following code samples:

Sample code for .NET - http://www.tmcSMS.com/api/tmc_API_dotnet.zip
Sample code for PHP - http://www.tmcSMS.com/api/tmc_API_php.zip
Sample code for Visual Basic - http://www.tmcSMS.com/api/tmc_API_vbs.zip
Sample code for Java - http://www.tmcSMS.com/api/tmc_API_java.zip

Please note that any language that supports SOAP or is able to call a URL can use the API. Should you require any assistance please contact support using the contact details below.

Contact details

The TMC support team are standing by to deal with any enquiries you may have and assist you with using our API to integrate the TMC Gateway with your systems. TMC Support can be contacted by calling us on + 44 207 224 4331 or emailing us at support@tmcSMS.com.

For corporate information please see www.tmcSMS.com.

Other message sending options, tools and details

The TMC SMS Service can also be accessed in the following ways:

1. Through our application website – www.tmcSMS.co.uk
2. Using our Desktop alerts program - <http://www.tmcSMS.co.uk/desktopicon.aspx>

For more details on The Message Centre, our services, pricing and other options please see www.tmcsm.com.

Troubleshooting

If you are having a problem sending messages the first thing to do is check the following items:

Issue	Suggestions
The Correct URL is being used	Check the HTTP URL that is being used in your request.
You are receiving invalid username even though you pass a valid username and password to the system	The system responds with invalid username if either the username or password is invalid. For all functions except Hash Password the password should be the result of the call to Hash Password and not produced via another hashing function. For Hash Password you should supply your plaintext password. You can also try logging into www.tmcsm.com to check the username and password are valid.
There are no network connection issues	Try the URL in your browser to ensure that the TMC API is reachable from the computer that you are trying to access it from. If not then check your broadband and network settings to ensure you are connected to the internet, check with your ISP that they have no issues and contact TMC Support for further assistance.
Your account is active, valid and you have credit	Verify that your account is active, in credit and ready to send messages using the Check Login method – see the Check Login section above for more details.
You have correctly hashed the password being used	The password passed to all functions must be hashed using the Hash Password method (with the exception of the Hash Password method itself). See the Hash Password section above for more details.
A Blank, Null or 0 is returned	This means that an error has occurred – follow the checks above to ensure that the TMC site is reachable and your account is active, valid and in credit. If the Check Login method call succeeded then please see the following suggestion for additional suggestions.
Login is fine but there is still an error with Blank, Null or 0 returned	Check that you are passing the hashed password (see above), next check the values you are passing (in particular the target phone number, parameters passed and parameter order). All parameters are required and must be present.
When sending messages via HTTP GET there is an error but the same url works fine in a web browser	Check that all parameters to the query string are URL encoded as some HTTP functions do not do this automatically and it can cause this problem. Also check that you have passed all parameters to the required function as defined above.

Characters are missing from the message when it reaches the destination	Check that you are using UTF-8 encoding as well as URL encoding – this is required for certain characters regardless of whether you set Unicode to true.
An error 500, Internal server error is returned	This can be due to a variety of reasons including valid parameters, syntax or combinations – please check all of your values and also check that you are using the correct options – e.g. Unicode=true if the message is Unicode.

If you are still having a problem then please contact TMC Support by emailing support@tmcsms.com or using the contact details section above.

Glossary

Term	Description
TMC	The Messaging Centre – a leading provider of SMS and other messaging services. See www.tmcsms.com for more details.
API	Application Programming Interface – a system for integrating
URL	Universal Resource Locator – a unique address of an internet resource with optional parameters to pass to that resource.
Inbound	When messages are sent from other providers (such as mobile networks) into our messaging system.
Outbound	When messages are sent out from our messaging system to other providers (such as mobile networks)
Hash	An encoded value that is always returned from the same given input to a Hashing function – the Hash cannot be reversed to find the original value providing additional security.
HTTP	Hyper Text Transfer Protocol – the primary method of transferring content over the internet.
SOAP	Simple Object Access Protocol – An XML based standard that allows for involving of web service and other methods from remote systems.
WAP Push	A SMS text message that contains a selectable URL that takes the user to a target web page in the browser or other software.