

Digitaleo

BUSINESS BOOSTER

API SMS

Revisions

Revision	Author	Date	Comments
1	pmarechal	21/06/2016	English version

Table des matières

1. API SETTINGS

- [1.1. TEXT MESSAGE](#)
- [1.2. LONG TEXT MESSAGE](#)
- [1.3. BINARY TEXT MESSAGE](#)
- [1.4. INTERNATIONAL TEXT MESSAGES](#)
- [1.5. MOBILE NUMBERS](#)
- [1.6. CODE](#)
- [1.7. SCHEDULED TEXT MESSAGES](#)
- [1.8. VOLUME](#)
- [1.9. TEXT MESSAGE STATUSES](#)
- [1.10. RETURN](#)
- [1.11. LIMITATIONS](#)
- [1.12. RESPONSES](#)
- [1.13. PING](#)
- [1.14. OPTIONAL PARAMETERS](#)
- [1.15. PHP EXAMPLES](#)

2. SOAP API

- [2.1. WSDL ADDRESS](#)
- [2.2. SOAP METHODS](#)
- [2.3. SENDING A TEXT MESSAGE](#)
- [2.4. SENDING A SIMPLE TEXT MESSAGE MAILING](#)
- [2.5. SENDING A PERSONALIZED TEXT MESSAGE MAILING](#)
- [2.6. SENDING A PERSONALIZED TEXT MESSAGE MAILING \(advanced mode\)](#)
- [2.7. TEXT MESSAGE STATUSES](#)
- [2.8. STATUSES OF TEXT MESSAGE MAILINGS](#)
- [2.9. STATUSES OF TEXT MESSAGES WHOSE STATUSES HAS CHANGES SINCE THE LAST STATUSES REQUEST](#)
- [2.10. CANCELING A SMS](#)
- [2.11. CHECKING A MOBILE PHONE NUMBER](#)

3. HTTP API

- [3.1. HTTP ADDRESS](#)
- [3.2. HTML RETURN](#)
- [3.3. SENDING A TEXT MESSAGE](#)
- [3.4. SENDING A SIMPLE TEXT MESSAGE MAILING](#)
- [3.5. TEXT MESSAGE STATUS](#)
- [3.6. STATUSES OF TEXT MESSAGE MAILINGS](#)
- [3.7. MOBILE NUMBER CHECKING](#)

4. SMTP API

- [4.1. E-MAIL ADDRESS](#)
- [4.2. SENDING A TEXT MESSAGE MAILING](#)

5. FTP API

[5.1. FTP ADDRESS](#)

[5.2. SENDING A TEXT MESSAGE MAILING](#)

[6. TEXTEO MONITOR WEBSITE](#)

[6.1. WEB ADDRESS](#)

[6.2. FORMAT](#)

[7. TEXTEO WEBSITE](#)

[7.1. WEB ADDRESS](#)

[7.2. FORMAT](#)

[A. APPENDIX A - TEXT MESSAGE CHARACTERS](#)

[B. APPENDIX B - SOAP METHODS](#)

[Copyright](#)

This document describes the interfaces, or APIs, used by Digitaleo's ecoSMS platform that you can use to send text messages, check their status, and read replies.

These are the various interfaces:

- **SOAP API: "web service"** interface (XML/WSDL) – the API we recommend
- **HTTP API: web** interface to send a small number of text messages – quick to implement
- **HTTPS API: secure web** interface, same functionally than HTTP API
- **SMTP API: e-mail** interface (mail2sms) to send high volumes of mailings
- **FTP API: file transfer interface**, same use as SMTP API

Each interface can provide all or some of the following services:

- **sending a text message:** to send a text message to a recipient
- **receiving a text reply:** to get the replies sent by your text message recipients
- **sending a simple text message mailing:** to send the same text message to a high number of recipients
- **sending a personalized text message mailing:** to send a personalized text message to a high number of recipients
- **text message status check:** to monitor the status of one or more text messages over time using its ID, the ID or name of a mailing, or the most recent messages
- **canceling a text message or mailing:** to cancel a scheduled text message or mailing before it is sent (or even while it is being sent, in the case of mailings)
- **number checking:** checks mobile phone numbers before sending
- **text message statistics:** to see at a glance how many text messages you send by month, by status

The table below presents the availability of services and options by API:

	Services							Options			
	Sending a text message	Simple text mailing	Personalized text mailing	Text message status	Replies	Text message statistics	Number checking	Canceling text messages or mailings	Binary text message	Long text message	Scheduled text message
SOAP API	X	X	X	X	X		X	X	X	X	X
HTTP API	X	X		X	X		X	X		X	X
HTTPS API	X	X		X	X		X	X		X	X
SMTP API	X	X	X							X	
FTP API	X	X	X							X	

1. API SETTINGS

1.1. TEXT MESSAGE

Text messages are limited to 160 characters.

The following 10 characters are counted twice: ^\|~[]}, the euro symbol (€), and the Form Feed / FF character (0x0C).

See appendix A for a list of characters supported in text mode.

1.2. LONG TEXT MESSAGE

Can be used to send longer text messages, up to 1024 characters.

Long text messages are broken up into several short text messages, then reassembled on the user's mobile phone. Each text message contains 153 characters at most (the characters that are counted twice in short text messages are also counted twice here). If the mobile phone does not support long text messages, the message will appear as several messages.

The cost of a long text message is the sum of the cost of each short text message.

To activate this option for your account, you must make an explicit request. Text messages are limited to 160 characters by default.

1.3. BINARY TEXT MESSAGE

Binary mode lets you use any character in your text message. For example, you can use Chinese characters if you want to send a text message to China. A binary text message is composed of 70 characters.

1.4. INTERNATIONAL TEXT MESSAGES

Can be used to send international text messages.

You must explicitly request that the "international" option be activated for your account. If you don't, text messages sent to a foreign mobile phone will not be delivered.

1.5. MOBILE NUMBERS

Mobile numbers must comply with the international format (e.g. +44 71234 56789)

The international country code for your country is defined when your Texteo account is created ("44" by default for the United Kingdom).

1.6. CODE

A code is required to access the various services.

Contact us to set up a trial account.

1.7. SCHEDULED TEXT MESSAGES

You can schedule a text message to be sent at a later time. If no date is specified, the message will be sent immediately.

1.8. VOLUME

- **HTTP or HTTPS API:** lets you send up to 200 text messages at once
- **SOAP, SMTP, or FTP API:** lets you send up to 2,000 text messages at once (contact us if you need to send more than 2,000 messages at once)
- **Check most recent statuses:** up to 2,000

Note: text messages are sent at an average rate of 20 per second.

1.9. TEXT MESSAGE STATUSES

The possible statuses of a text message are outlined in the table below:

Website indication	API code returned	Description
Programmé	wait	The text message has been scheduled but not yet sent.
En cours	on	The text message has been sent but the recipient has not yet received it.
En échec	ko	The text message could not be sent. The reason for failure is given.
Délivré	ok	The text message has been delivered to its recipient.
Répondu	rep	The text message has been delivered and the recipient has sent a reply.
Inconnu	no	The final status of the text message is unknown.

In addition, a more specific technical code like "N5" or "N4:204" format is returned along with a more detailed description. These additional elements may change over time and are given for information purposes only.

1.10. RETURN

For each of these functions, a return is sent back according to the following settings:

for each text message sent

- **sms_status** (string): the text message status (old format)
- **sms_id** (int): the text message ID number
- **status** (statusType): the text message status (new format). See ecoSMS statusType WSDL definition for the detailed structure.
- **mobile** (string): the standardized number of the mobile phone associated with the status
- **date** (dateTime): date of the last return
- **reponse** (string): reply if relevant
- **parts** (int): the number of short text messages sent

for a mailing

- **mailing_id** (int): the mailing ID number

1.11. LIMITATIONS

The status check frequency is limited. The limit is not currently categorically frozen but a moderate usage would be not to call more than 10 status requests per SMS sent.

So, it's also recommended:

- to group the status check into one request
- to use the status check of a mailing that returns the whole statuses of a mailing with only one request.
- to use the last_n parameter
- not to check the status of an SMS that has already a final status (that cannot change)

If an abusive use is noticed, the status check could be temporarily disabled.

With one request, you can get up to 2,000 statuses.

1.12. RESPONSES

As indicated above, replies to text messages are returned as a status.

In addition, responses can be sent to an e-mail address specific to your account. The body of the e-mail message contains the text message that was sent along with its response, and the subject of the e-mail is "Response from 801 555 xxxx to SMS #xxx," where "801 555 xxxx" is the sender's mobile number and "#xxx" is the ID number of the corresponding text message.

1.13. PING

Instead of constantly requesting the status of a text message until it is received (polling), you can be alerted when status changes or when a reply is submitted to text messages you have sent.

For this, you provide an address on your server, e.g. <http://www.serveur.com/ping.asp>, and the Texteo platform will call this address with the text message ID each time there is an update, such as: http://www.serveur.com/ping.asp?sms_id=xxxx

1.14. OPTIONAL PARAMETERS

Over the years, some optional parameters have been added to new API versions to clarify the use of the services.

Only for the SOAP API, in order to keep backward compatibility, an optional parameter can be simply missing or set with attribute nil="true"

Some optional parameters are not documented, they relate to special use cases of the API.

1.15. PHP EXAMPLES

PHP examples are included in this guide to illustrate the services available.

These examples require the use of PHP 5.x or later built with the SOAP support.

2. SOAP API

The SOAP API is a “web service” interface. It is the best interface for sending text messages through the Texteo platform.

2.1. WSDL ADDRESS

The WSDL address of the ecoSMS SOAP service is: <http://www.ecosms.fr/ecosms.wsdl>

2.2. SOAP METHODS

This SOAP API offers the following methods:

- smsSend: to send a text message or a simple text mailing
- smsSendMultiple: to send a personalized text message mailing
- smsStatus: to recover the status and any replies to one or more text messages
- mobilesCheck: to check mobile numbers

The services described below all use one of these three methods.

2.3. SENDING A TEXT MESSAGE

A text message sent to a mobile phone

smsSend service

settings:

- **sms** (string): the text message to be sent
- **mobile** (string): the mobile phone number
- **code** (string): your access code

optional settings:

- **pays** (string): country of the mobile number in domestic format
- **date** (dateTime): scheduled delivery date
- **bin** (boolean): binary or text mode

PHP example

This is the generic PHP code to use for sending text messages through the SOAP API:

```
<?php

$ecoSmsSoapClient=new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );
$result=$ecoSmsSoapClient->smsSend( ... settings in table... );
```

To send a text message through the SOAP API:

```
<?php

$result=$ecoSmsSoapClient->smsSend(
    array(
        'sms'=>'test message',
        'mobile'=>'+447123456789',
        'code'=>'DEMO' ) );
```

To send a text message in Belgium:

```
<?php

$result=$ecoSmsSoapClient->smsSend(
    array(
        'sms'=>'test message',
        'mobile'=>'+320412345678',
        'country'=>'BE',
        'code'=>'DEMO' ) );
```

To send a text message 10 minutes from now:

```
<?php

$date = date( 'c', time() + 600 ); // ISO format 8601 for dateTime required by SOAP
$result=$ecoSmsSoapClient->smsSend(
array(
    'sms'=>'test message',
    'mobile'=>'+447123456789',
    'date'=>$date,
    'code'=>'DEMO' ) );
```

To send a binary text message:

```
<?php

$result=$ecoSmsSoapClient->smsSend(
array(
    'sms'=>'世界您好',
    'bin'=>true,
    'mobile'=>'+861312345678',
    'code'=>'DEMO' ) );
```

To display the result of the send, add these lines:

```
<?php

echo "sms_id={$result->smsStatus->sms_id}<br \>";
echo "statut={$result->smsStatus->sms_status}<br \>";
echo "mobile={$result->smsStatus->mobile}<br \>";
echo "date={$result->smsStatus->date}<br \>";
```

2.4. SENDING A SIMPLE TEXT MESSAGE MAILING

A text message sent to several mobile phones

smsSend service

settings:

- **sms** (string): the text message to be sent
- **mobile** (string): the mobile phone number
- **code** (string): your access code

optional settings:

- **pays** (string): country of the mobile number in domestic format
- **date** (dateTime): scheduled delivery date
- **nom** (string): name given to this mailing
- **bin** (boolean): binary or text mode

PHP example

This is the PHP code to send a simple text message mailing through the SOAP API:

```
<?php
$ecoSmsSoapClient = new SoapClient("http://www.ecosms.fr/ecosms.wsdl");
$mobiles = array( '+447123456789', '+447123456789', '+447123456789' );
$result = $ecoSmsSoapClient->smsSend(
    array(
        'sms'=>'test message',
        'nom'='test mailing',
        'mobile'=>$mobiles,
        'code'=>'DEMO' ));
```

To display the result of the mailing sent, add these lines:

```
<?php
echo "mailing_id={$result->mailing_id}<br \><br \>";
foreach($result->smsStatus as $status)
{
    echo "sms_id={$status->sms_id}<br \>";
    echo "statut={$status->sms_status}<br \><br \>";
    echo "mobile={$status->mobile}<br \>";
    echo "date={$status->date}<br \>";
}
```

2.5. SENDING A PERSONALIZED TEXT MESSAGE MAILING

A personalized text message sent to several mobile phones

smsSend service

settings:

- **code** (string): your access code
- one or more occurrences of the two following settings:*
- **sms** (string): the text message to be sent
 - **mobile** (string): the mobile phone number

optional settings:

- **pays** (string): country of the mobile number in domestic format
- **date** (dateTime): scheduled delivery date
- **nom** (string): name given to this mailing
- **bin** (boolean): binary or text mode

PHP example

This is the PHP code to send a personalized text message mailing through the SOAP API:

```
<?php
$ecoSmsSoapClient=new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );

$smsMultiple[] = array(
    'sms' => 'Dear Mr. Smith, your car is ready',
    'mobile' => '+447123456789');
$smsMultiple[] = array(
    'sms' => 'Dear Mrs. Doe, your car is ready',
    'mobile' => '+447123456789');

$result=$ecoSmsSoapClient->smsSendMultiple(
    array('smsMultiple' => $smsMultiple, 'code' => 'DEMO') );
```

2.6. SENDING A PERSONALIZED TEXT MESSAGE MAILING (advanced mode)

This advanced mode can be used to send personalized text messages to several contacts.

A contact is associated with a mobile number to which fields are attributed, such as "last name" and "first name." It is then possible to include these fields in the text message by enclosing them between pound signs (#).

Sample text message: "Dear #FIRSTNAME# #LASTNAME#, we would like to inform you..."

smsSend service

settings:

- **code** (string): your access code

one or more occurrences of the two following settings:

- **sms** (string): the text message to be sent
- **contact**: description of contacts (see ecoSMS smsSend WSDL definition for the detailed structure)

optional settings:

- **pays** (string): country of the mobile number in domestic format
- **date** (dateTime): scheduled delivery date
- **nom** (string): name given to this mailing
- **bin** (boolean): binary or text mode

PHP example

This is the PHP code to send a personalized text message mailing through the SOAP API:

```
<?php
$ecoSmsSoapClient=new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );

$contact1['mobile'] = '+447123456789';
$contact1['param'][] = array( 'nom' => 'LASTNAME', '_' => "Smith" );
$contact1['param'][] = array( 'nom' => 'FIRSTNAME', '_' => "Peter" );
$contacts[]=$contact1;

$contact2['mobile'] = '+447123456789';
$contact2['param'][] = array( 'nom' => 'LASTNAME', '_' => "Dupont" );
$contact2['param'][] = array( 'nom' => 'FIRSTNAME', '_' => "Jack" );
$contacts[]=$contact2;
```



```
$sms = 'Dear #FIRSTNAME# #LASTNAME#, you are cordially invited to a cocktail reception to  
unveil our new vehicle';
```

```
$result=$ecoSmsSoapClient->smsSend(  
    array(  
        'sms'=>$sms,  
        'contact'=>$contacts,  
        'code'=>'CODE'  
    )  
);
```

2.7. TEXT MESSAGE STATUSES

Reading the status of one or more text messages

smsStatus service

one or more occurrences of the two following settings:

- **sms_id** (int): the text message ID number

or:

- **last_n** (int): number of statuses requested (1,000 max.)

then this setting:

- **code** (string): your access code

PHP example

This is the PHP code used to read and display the status of a text message:

```
<?php

$ecoSmsSoapClient = new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );
$result = $ecoSmsSoapClient->smsStatus( array( 'sms_id'=>'2382936', 'code' => 'DEMO' ) );
echo "Statut = " . $result->smsStatus->sms_status ;
```

This is the PHP code used to read and display the status of several text messages:

```
<?php

$ecoSmsSoapClient = new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );

$sms_ids = array( 2537774, 2537775, 2537776 );
$result = $ecoSmsSoapClient->smsStatus( array( 'sms_id'=>$sms_ids, 'code' => 'DEMO' ) );

foreach($result->smsStatus as $status)
{
    echo "sms_id={$status->sms_id}<br \>";
    echo "mobile={$status->mobile}<br \>";
    echo "statut={$status->sms_status}<br \><br \>";
    echo "date={$status->date}<br \>";
    echo "rep={$status->reponse}<br \>";
}
}
```

2.8. STATUSES OF TEXT MESSAGE MAILINGS

Reading the status of one or more text message mailings

smsStatus service

one or more occurrences of the two following settings:

- **mailing_id** (int): the text message mailing ID number

or:

- **nom** (string): text message mailing name

then this setting:

- **code** (string): your access code

PHP example

This is the PHP code used to read and display the status of a text message mailing:

```
<?php

$ecoSmsSoapClient = new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );
$result = $ecoSmsSoapClient->smsStatus(
    array(
        'Mailing_id' => 254,
        'code'       => 'DEMO'
    )
);

foreach($result->smsStatus as $status)
{
    echo "sms_id={$status->sms_id}<br \>";
    echo "statut={$status->sms_status}<br \><br \>";
    echo "mobile={$status->mobile}<br \>";
    echo "date={$status->date}<br \>";
}
```

2.9. STATUSES OF TEXT MESSAGES WHOSE STATUSES HAS CHANGES SINCE THE LAST STATUSES REQUEST

This services aims to read only text messages statuses whoses status has changed since the last request. A read text message during the SOAP request "k" will not be returned during the request "k+1" and next requests only if its status is changed.

smsStatusQueue service

Parameters

- **code** (string): your access code
- **n** (int): Maximum number of returned text messages statuses.

Return: smsStatus

Php example

This is the PHP code used to read and display the status of the last 10 changed text message mailing:

```
<?php
$ecoSmsSoapClient = new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );
$result = $ecoSmsSoapClient->smsStatusQueue( array( 'n'=>10, 'code' => 'DEMO' ) );
foreach($result->smsStatus as $status)
{
    echo "sms_id={$status->sms_id}<br \>";
    echo "statut={$status->sms_status}<br \><br \>";
    echo "mobile={$status->mobile}<br \>";
    echo "date={$status->date}<br \>";
}
```

2.10. CANCELING A SMS

Canceling one or more text messages

smsStatus service

settings:

- **code** (string): your access code
- **stop** (Boolean): cancel mode

one or more occurrences of the following settings:

- **sms_id** (int): the text message ID number
- **mailing_id** (int): the text message mailing ID number

PHP example

This is the PHP code used to read, display the status and cancel several text messages:

```
<?php

$ecoSmsSoapClient = new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );

$sms_ids = array( 2537774, 2537775, 2537776 );
$result = $ecoSmsSoapClient->smsStatus(
    array(
        'Sms_id' => $sms_ids,
        'code'   => 'DEMO',
        'stop'   => 'true'
    )
);

foreach($result->smsStatus as $status)
{
    echo "sms_id={$status->sms_id}<br \>";
    echo "mobile={$status->mobile}<br \>";
    echo "statut={$status->sms_status}<br \><br \>";
    echo "date={$status->date}<br \>";
    echo "rep={$status->reponse}<br \>";
}
}
```

2.11. CHECKING A MOBILE PHONE NUMBER

mobileCheck service

settings:

- **code** (string): your access code

one or more occurrences of the following settings:

- **mobile** (string): mobile phone number

Return:

one or more occurrences of the following parameters:

- **check** (string): ok if the mobile phone number is valid, ko otherwise
- **prefix** (string): international code of the mobile phone number
- **international** (string): international format of the mobile phone number

PHP example

This is the PHP code used to read one or more mobile phone number and display them:

```
<?php

$ecoSmsSoapClient = new SoapClient( "http://www.ecosms.fr/ecosms.wsdl" );
$result = $ecoSmsSoapClient->mobilesCheck(
    array(
        'mobile'=>'+447123456789',
        'code' => 'DEMO' ) );

foreach($result->mobile as $mobile)
{
    echo "check={$mobile->check}<br \>";
    echo "prefix={$mobile->prefix}<br \><br \>";
    echo "international={$mobile->international}<br \>";
}
```

3. HTTP API

3.1. HTTP ADDRESS

This is the internet link to the:

- HTTP API: <http://www.ecosms.fr/ecosms.php>
- HTTPS API: <https://www.ecosms.fr/ecosms.php>

This service can be called up manually using a web browser (Internet Explorer, Firefox, etc.) or it can be programmed. Below you will find examples programmed in PHP.

When included in the address, the setting text must be encoded using RFC 1738 rules (among others, these include replacing spaces with %20). Browsers do this automatically; a specific function is present in most languages ("rawurlencode" in PHP, for example).

Dates are encoded according to ISO standard 8601 in this format: "2007-12-10T13:46+01:00."

3.2. HTML RETURN

The return is in text format (text/plain) and is similar for all queries described below, in this form:

```
mailing_id=nnn (optionnal)

sms_id=nnn
statut=ok (N5:SMS received)
mobile=447123456789
date=2007-12-10T13:45+01:00

sms_id=nnn
statut=ok (N5:SMS received)
mobile=447123456789
date=2007-12-10T13:46+01:00

sms_id=nnn
statut=ok (N5:SMS received)
mobile=447123456789
date=2007-12-10T13:46+01:00
```

3.3. SENDING A TEXT MESSAGE

The general syntax of the URL to send a text message is as follows:

```
http://www.ecosms.fr/ecosms.php?sms=message&mobile=447123456789&code=DEMO&date=2007-12-10T13:46
```

where:

- **sms:** the text message to be sent
- **mobile:** the mobile phone number
- **code:** your access code
- **date:** scheduled sending date (optional setting)

PHP example

This is the PHP code to send a text message through the HTTP API:

```
<?php
$sms    = rawurlencode('test message');
$mobile = '447123456789';
$code   = 'DEMO';
$date   = '2007-12-11T15:58';

$ecoSmsAdresse =
"http://www.ecosms.fr/ecosms.php?sms={$sms}&mobile={$mobile}&code={$code}&date={$date}";

header( "Content-Type: text/plain" );
readfile ($ecoSmsAdresse);
```


3.4. SENDING A SIMPLE TEXT MESSAGE MAILING

To send a simple text message mailing, replace the mobile phone number with the list of recipient mobile numbers separated with dashes (-). For example, replace

447123456789 with 447123456789-441234567891-441234567892

The general syntax of the URL to send a simple text message mailing is as follows:

```
http://www.ecosms.fr/ecosms.php?sms=votre message
&code=DEMO&mobile=447123456789-441234567891-441234567892
```

PHP example

This is the PHP code to send a simple text message mailing through the HTTP API:

```
<?php

$sms      = rawurlencode('test message');
$mobile  = '447123456789-441234567891-441234567892';
$code    = 'DEMO';

$ecoSmsAdresse = "http://www.ecosms.fr/ecosms.php?sms={$sms}&mobile={$mobile}&code={$code}";

header( "Content-Type: text/plain" );
readfile ($ecoSmsAdresse);
```

3.5. TEXT MESSAGE STATUS

The syntax to monitor the status of a text message is as follows:

```
http://www.ecosms.fr/ecosms.php?sms_id=235262-235274-235265&code=DEMO
```

where

- **code**: your access code
- **sms_id**: the text message ID numbers separated by dashes (-)

PHP example

This is the PHP code used to read and display the status of one or more text messages through the HTTP API:

```
<?php
$sms_ids = '235262-235274-235265';
$code    = 'DEMO';

$ecoSmsAdresse = "http://www.ecosms.fr/ecosms.php?sms_id={$sms_ids}&code={$code}";

header( "Content-Type: text/plain" );
readfile($ecoSmsAdresse);
```

3.6. STATUSES OF TEXT MESSAGE MAILINGS

The syntax to monitor the status of a text message mailing is as follows:

```
http://www.ecosms.fr/ecosms.php?mailing_id=237682
```

where

- **code**: your access code
- **mailing_id**: the text message ID number(s) separated by dashes (-)

PHP example

This is the PHP code used to read the status of mailing text messages through the HTTP API:

```
<?php
$mailing_id = '12';
$code      = 'DEMO';

$ecoSmsAdresse = "http://www.ecosms.fr/ecosms.php?mailing_id={$mailing_id}&code={$code}";

header( "Content-Type: text/plain" );
readfile($ecoSmsAdresse);
```

3.7. MOBILE NUMBER CHECKING

The general syntax of the URL to check numbers is as follows:

```
http://www.ecosms.fr/ecosms.php?code=DEMO&mobile=447123456789-441234567891-4471234567892
```

where

- **code** (string): your access code

one or more occurrences of the following setting:

- **mobile** (string): mobile phone number

The return is in this format:

```
mobile=447123456789
check=ko
prefix=-1
international=

mobile=324612345679
check=ok
prefix=32
international=+32412345679

mobile=447123456789
check=ko
prefix=-1
international=
```

Where **mobile** contains the number requested, **check** indicates whether the number is valid (ok) or not (ko), **international** shows the number in international format if it is valid, **prefix** gives the international prefix if the number is valid (otherwise empty or negative if it "appears" to be a number with this country code).

4. SMTP API

4.1. E-MAIL ADDRESS

To send your text message mailings, use the following SMTP API e-mail address:

```
sms@ecosms.fr
```

4.2. SENDING A TEXT MESSAGE MAILING

To send a text message mailing, just send an e-mail to sms@ecosms.fr with the following text (either in the message body or in an attachment):

```
U-code  
T-first text message for 3 recipients  
#-447123456789  
#-441234567891  
#-441234567892  
T-second text message  
T-for 2 recipients  
#-441234567893  
#-441234567894  
T-last text message for 3  
#-441234567895  
#-441234567896  
#-441234567897
```

Where:

- a line beginning with U- containing the "code" of your texting account
- one or more lines beginning with T- containing the text message (160 characters at most) to be sent to the recipients listed underneath
- one recipient per line beginning with #- (unlimited recipients)
- to send different text messages in a single e-mail, just alternate the messages in the text message fields (those that begin with "T-") and the recipients' mobile numbers (fields beginning with "#-"). Please note that the message is sent to the mobile numbers listed AFTER the message text.
- The subject of the e-mail and the name of the attachment do not matter.

PHP example

This is the PHP code to send a text message mailing through the SMTP API:

```
<?php
$body = 'U-DEMO
T-first text message for 3 recipients
#-447123456789
#-441234567891
#-441234567892
T-second text message
T-for 2 recipients
#-441234567893
#-441234567894
T-last text message for 4
#-441234567895
#-441234567896
#-441234567897
#-441234567898;

mail( 'sms@ecosms.fr', "Send text message", $body );
```

5. FTP API

5.1. FTP ADDRESS

This is the address of the FTP API server:

```
ftp://ftp.ecosms.fr
```

5.2. SENDING A TEXT MESSAGE MAILING

Contact us to obtain a username and password.

The format for files to be sent to the server is the same as for the SMTP API. See the previous section for details.

6. TEXTEO MONITOR WEBSITE

6.1. WEB ADDRESS

This is the address of the Texteo Monitor website:

```
https://monitor.texteo.net/
```

6.2. FORMAT

The Texteo Monitor website lets you lets you monitor text messages sent by the following ecoSMS APIs: WSDL, HTTP, HTTPS, SMTP, and FTP.

Contact us to set up an account.

7. TEXTEO WEBSITE

7.1. WEB ADDRESS

This is the address of the Texteo website:

```
http://www.texteo.net
```

7.2. FORMAT

The Texteo website is a complete web application for sending professional text messages. It can be used to send text messages and mailings according to predefined, business-specific models and personalized address books.

Contact us to set up an account.

A. APPENDIX A - TEXT MESSAGE CHARACTERS

List of supported characters for text messages in text mode

1. Single characters

@	Δ	<SP>	0	i	P	ı	p
£	_	!	1	A	Q	a	q
\$	Φ	"	2	B	R	b	r
¥	Γ	#	3	C	S	c	s
è	Λ	¤	4	D	T	d	t
é	Ω	%	5	E	U	e	u
ù	Π	&	6	F	V	f	v
ì	Ψ	'	7	G	W	g	w
ò	Σ	(8	H	X	h	x
ç	Θ)	9	I	Y	i	y
<LF>	Ξ	*	:	J	Z	j	z
Ø	<ESC>	+	;	K	Ä	k	ä
ø	Æ	,	<	L	Ö	l	ö
<CR>	æ	-	=	M	Ñ	m	ñ
À	ß	.	>	N	Ü	n	ü
â	É	/	?	O	Ş	o	à

2. Double characters

€ <FF> [\]

^ { | } ~

Reference for the GSM 03.38 character set:

<http://www.unicode.org/Public/MAPPINGS/ETSI/GSM0338.TXT>

The following characters are changed when the message is sent:

- a with circumflex 'Â' => 'A' , 'â' => 'a'
- e with circumflex 'Ê' => 'E' , 'ê' => 'e'
- i with circumflex 'Î' => 'I' , 'î' => 'i'
- o with circumflex 'Ô' => 'O' , 'ô' => 'o'
- u with circumflex 'Û' => 'U' , 'û' => 'u'
- e trema 'Ë' => 'E' , 'ë' => 'e'
- i trema 'Ï' => 'I' , 'ï' => 'i'
- Degree symbol '°' => 'o'
- Ligatured Æ 'œ' => 'oe'
- grave accent '`' => ''
- smart apostrophe ''' => ''
- ellipsis points '...' => '...'

Characters not listed here are deleted.

B. APPENDIX B - SOAP METHODS

The SOAP API provides the following methods:

smsSend: Sending a simple text message or a simple text message mailing

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ecos="http://www.ecosms.fr">
  <soapenv:Header/>
  <soapenv:Body>
    <ecos:smsSend>
      <smsSendRequest>
        <sms>?</sms>
        <bin>?</bin>
        <!--You have a CHOICE of the next 2 items at this level-->
        <!--1 or more repetitions:-->
        <mobile>?</mobile>
        <!--1 or more repetitions:-->
        <contact>
          <mobile>?</mobile>
          <!--1 or more repetitions:-->
          <param nom="?">?</param>
        </contact>
        <code>?</code>
        <pays>?</pays>
        <tpoa>?</tpoa>
        <nom>?</nom>
        <date>?</date>
        <type>?</type>
      </smsSendRequest>
    </ecos:smsSend>
  </soapenv:Body>
</soapenv:Envelope>
```

smsSendMultiple: Sending a personalized text message mailing

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
                    xmlns:ecos="http://www.ecosms.fr">
  <soapenv:Header/>
  <soapenv:Body>
    <ecos:smsSendMultiple>
      <smsSendMultipleRequest>
        <!--1 or more repetitions:-->
        <smsMultiple>
          <sms>?</sms>
          <bin>?</bin>
          <!--You have a CHOICE of the next 2 items at this level-->
          <!--1 or more repetitions:-->
          <mobile>?</mobile>
          <!--1 or more repetitions:-->
          <contact>
            <mobile>?</mobile>
            <!--1 or more repetitions:-->
            <param nom="?">?</param>
          </contact>
        </smsMultiple>
        <code>?</code>
        <pays>?</pays>
        <tpoa>?</tpoa>
        <nom>?</nom>
        <date>?</date>
        <type>?</type>
      </smsSendMultipleRequest>
    </ecos:smsSendMultiple>
  </soapenv:Body>
</soapenv:Envelope>
```

smsStatus: Reading the status of one or more text messages, canceling one or more text messages

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ecos="http://www.ecosms.fr">
  <soapenv:Header/>
  <soapenv:Body>
    <ecos:smsStatus>
      <smsStatusRequest>
        <!--You have a CHOICE of the next 4 items at this level-->
        <!--1 or more repetitions:-->
        <sms_id?</sms_id>
        <!--1 or more repetitions:-->
        <mailing_id?</mailing_id>
        <nom?</nom>
        <last_n?</last_n>
        <stop?</stop>
        <code?</code>
      </smsStatusRequest>
    </ecos:smsStatus>
  </soapenv:Body>
</soapenv:Envelope>
```

mobilesCheck: number checking: checks mobile phone numbers before sending

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ecos="http://www.ecosms.fr">
  <soapenv:Header/>
  <soapenv:Body>
    <ecos:mobilesCheck>
      <mobilesCheckRequest>
        <!--1 or more repetitions:-->
        <mobile?</mobile>
        <code?</code>
      </mobilesCheckRequest>
    </ecos:mobilesCheck>
  </soapenv:Body>
</soapenv:Envelope>
```

Services described in this document all use one of these methods.

Copyright

All of this code is governed by French and international legislation on copyright and intellectual property. All reproduction rights reserved, including for documents that can be downloaded and iconographic and photographic representations. Reproducing all or a portion of this code on any support whatsoever is strictly forbidden unless authorization is obtained in writing from Digitaleo.