



## **Integrating with Cellsynt's SMS gateway via HTTP interface (technical documentation)**



# Table of Contents

<b>Part I Introduction</b>	<b>1</b>
1 Features .....	1
2 Minimum requirements .....	1
3 Support .....	1
<b>Part II Quick start: Example flow</b>	<b>1</b>
<b>Part III Send SMS</b>	<b>2</b>
1 Parameters .....	2
2 Message types .....	3
Text .....	4
Long SMS .....	4
Binary .....	4
Unicode .....	5
3 Return value .....	5
Successful request .....	5
Failed request .....	5
Out of credit.....	6
<b>Part IV Delivery reports</b>	<b>6</b>



# 1 Introduction

This is a technical guide describing how to send SMS and receive delivery reports by integrating with Cellsynt's SMS gateway using a simple HTTP interface. The interface has been developed to provide a quick and easy way to get started with SMS using technologies and knowledge most developers possess.

Cellsynt's SMS gateway is a reliable, flexible and scalable gateway which in an easy way enables development of simple as well as advanced SMS services supporting all kinds of message types.

## 1.1 Features

- Communication via a simple HTTP interface (quick and easy to get started)
- Possibility to send SMS to multiple recipients in a single HTTP request
- Support for all types of messages
- Delivery reports

## 1.2 Minimum requirements

- Programming knowledge in a suitable programming language which is able to handle HTTP communication (e.g. ASP.NET, Java, Perl, PHP, Python, Ruby or similar)
- Delivery reports require a public webserver available over the Internet to receive the reports via HTTP communication

## 1.3 Support

If questions or problems arise which cannot be answered by this manual please e-mail us on [support@cellsynt.com](mailto:support@cellsynt.com) or call our office at +46 8 525 09 700.

To help us provide you with fast and accurate service please include as many details as possible, for example:

- Account (username)
- Detailed description of issue
- Time it occurred (together with time zone)
- Message originator and destination(s)
- Possible error messages/logs

# 2 Quick start: Example flow

1. A message is sent to UK mobile number 07920-110 000 with the text "Testing 123" and originator "Demo". The following HTTP GET request is made:

```
https://se-1.cellsynt.net/sms.php?
username=demo&password=test123&destination=00447920110000&type=text&charset=UTF-
8&text=Testing%20123&originatortype=alpha&originator=Demo
```

2. The following unique reference (*trackingid*) is received from the web server's response value:  
OK: 92dff27302b754424242fb204620dc18

3. Delivery reports are configured for the account to URL <https://example.org/delivery-receipts>

The message gets delivered to the mobile phone and the following HTTP GET request is made:

```
https://example.org/delivery-receipts?
trackingid=92dff27302b754424242fb204620dc18&status=delivered&destination=0044792
0110000
```



### 3 Send SMS

Sending SMS is done by making a HTTP GET- or POST request. There is no difference in which method you choose but under certain circumstances, such as sending messages to larger amounts of numbers, HTTP POST is recommended because of technical limitations in the length of a HTTP GET query string.

HTTP requests should be made to the following URL:

<https://se-1.cellsynt.net/sms.php>

For compatibility reasons we also support requests via http, i.e. non-encrypted connections, but for your security we always recommend using https whenever possible.

The interface accepts parameters according to the table below. All parameter values should by default be supplied in character encoding ISO-8859-1 (Latin-1) but also UTF-8 is accepted if you specify it using the charset parameter.

#### 3.1 Parameters

Parameter	Mandatory	Description
username	X	Your username (received when account is setup).
password	X	Your password to use together with the username for authentication (received when account is setup).
destination	X	Recipient's telephone number on international format with leading 00 followed by country code, e.g. 00447920110000 for UK number 07920 110 000 (max 17 digits in total). To send the same message to multiple recipients, separate numbers with comma. Max 25000 recipients per HTTP request.
text	(X)	Text that should be sent. Max 160 characters (longer can be sent, see <a href="#">Long SMS</a> ) (Parameter <i>text</i> is mandatory as long as parameter <i>type</i> is not set. This is because <i>type</i> 's default value is <i>text</i> ).
charset		Character set text and other data is sent as in the HTTP request. Possible values: <i>ISO-8859-1</i> (default) and <i>UTF-8</i> .
originortype		Controls the originator type the message should be sent with. Possible values: <i>numeric</i> , <i>shortcode</i> and <i>alpha</i> .
originator		Identifier which will be visible on recipient's mobile phone as originator of the message. Allowed values and function depends on parameter <i>originortype</i> 's value according to below:  <u>numeric</u> Numeric value (max 15 digits) with telephone number on international format without leading 00 (example UK number 07920 110 000 should be set as 447920110000). Receiving mobile phone will add a leading + sign and thus see the originator as a normal mobile phone number (+447920110000). Therefore it is also possible to reply to the message.  <u>shortcode</u> Numerical value (max 15 digits). Used to set a shortcode in an operator network as originator (i.e. will be shown without leading + sign, e.g. 72456).  <u>alpha</u>



## Send SMS

		Alphanumeric string (max 11 characters). The following characters are guaranteed to work: a-z, A-Z and 0-9. Other characters <i>may</i> work but functionality can not be guaranteed. Recipients can not reply to messages with alphanumeric originators.
type		Type of message that should be sent. Possible values: <i>text</i> (default), <i>binary</i> and <i>unicode</i> . See more under <a href="#">Message types</a> .
allowconcat		Maximum number of SMS permitted to be linked together when needed (default value is 1, see <a href="#">Long SMS</a> ). Maximum value is 6 (i.e. max 153 x 6 = 918 characters).
expiry		Can be used if you want to prevent a message from being delivered after a certain time, e.g. 9 PM the same evening if information thereafter is considered invalid / outdated. If message has not been delivered after the set time (e.g. mobile phone was switched off) you will get a <a href="#">delivery receipt</a> with status "failed". Value should be given as a <a href="#">Unix timestamp</a> . Different operators permit different allowed max values (e.g. 3 days expiry time). If a value is set that is above an operator's max allowed time it will be adjusted to the highest possible value.
flash		Value can be set to true if message should be sent as "flash message", i.e. displayed directly on phone screen instead of being saved to inbox. This is identical to setting class=0. <i>Please note that support for flash messages cannot be guaranteed to all operator networks. If flash is not supported the message will be sent as a regular text message instead (class=1).</i>
class		Message class can be set to 0 (flash message), 1 (default, ME-specific), 2 (SIM-specific) or 3 (TE-specific).
udh		UDH (User Data Header) can be used to send concatenated SMS, contain formatting information, convey port numbers as a mean to cause start of an application etc. The value should be given on hexadecimal format for the corresponding bytes you wish to send (e.g. AABCC).
pid		Protocol Identifier (specified in GSM 03.40) says how the message should be interpreted. Value should be given on hexadecimal format, e.g. 00 for a regular message and 7D (decimal 125) for a configuration message ("ME Data download").

### 3.2 Message types

The most common message type is a normal text message which consists of up to 160 characters, but there are also other types of messages that can be sent. Via Cellsynt's SMS gateway the message type is controlled using the HTTP parameter *type*. Below follows a list of available types, areas of application, possible extra data needed and HTTP request examples.



### 3.2.1 Text

Used to send a normal text message. Maximum number of characters is 160. Any characters specified within [character coding GSM 03.38](#) can be used (e.g. English, Swedish, Norwegian), for other languages / alphabets (e.g. Arabic, Japanese, Chinese) please use [Unicode](#).

Example:

```
https://se-1.cellsynt.net/sms.php?
username=demo&password=test123&destination=00447920110000&type=text&charset=UTF-
8&text=Testing%20123
```

### 3.2.2 Long SMS

Long SMS (also known as "concatenated SMS") enables sending messages exceeding 160 characters. By sending several messages that apart from the text also contain information about how to assemble them again on the receiving mobile phone you can send longer messages that to the recipient looks like one single long message. The division and re-assembly of messages is taken care of by our platform and apart from the text you want to send you only need to specify one additional parameter: *allowconcat*.

Due to the additional information required for re-assembly each part in the long SMS can only contain 153 characters (instead of 160 characters). The numeric parameter *allowconcat* specifies the amount of parts a message is permitted to be split into. Max value is 6 and default value is 1 (long SMS not permitted).

If you want to permit long messages to be sent at all times you can set the value of *allowconcat* to 6 for all messages (this lets you send up to  $153 \times 6 = 918$  characters). If you try to send a message longer than 160 characters and *allowconcat* is set to 1 (default), the message will be rejected.

Each part of a long SMS will be billed as one SMS. This means that if you send a text consisting of 450 characters with *allowconcat* set to 6 the message will fit into 3 SMS parts (maximum length for 3 SMS is  $153 \times 3 = 459$  characters) and the cost will be the same as for 3 regular messages.

Example:

```
https://se-1.cellsynt.net/sms.php?
username=demo&password=test123&destination=00447920110000&type=text&allowconcat=
6&charset=UTF-8&text>Lorem%20ipsum%20dolor%20sit%20amet%2C%20consectetur%
20adipiscing%20elit%2C%20sed%20do%20eiusmod%20tempor%20incididunt%20ut%20labore
%20et%20dolore%20magna%20aliqua.%20Ut%20enim%20ad%20minim%20veniam%2C%20quis%
20nostrud%20exercitation%20ullamco%20laboris%20nisi%20ut%20aliquip%20ex%20ea%
20commodo%20consequat.%20Duis%20aute%20irure%20dolor%20in%20reprehenderit%20in%
20voluptate%20velit%20esse%20cillum%20dolore%20eu%20fugiat%20nulla%20pariatur.%
20Excepteur%20sint%20occaecat%20cupidatat%20non%20proident%2C%20sunt%20in%
20culpa%20qui%20officia%20deserunt%20mollit%20anim%20id%20est%20laborum.
```

### 3.2.3 Binary

Binary messages can be used to send settings, business cards etc. See relevant phone manufacturer's manual for further specification on message formats available.

Parameter *type* should be set to *binary* to signal that you want to send a binary message.

Two parameters are needed to send the binary data: *udh* and *data*.

It is not mandatory to specify both, however, at least one of them must be set.

Parameter values should be specified as hexadecimal values. Both parameters must not together exceed 140 bytes = 280 characters.



Example:

```
https://se-1.cellsynt.net/sms.php?
username=demo&password=test123&destination=00447920110000&type=binary&udh=AABBCC
001122&data=334455FF
```

### 3.2.4 Unicode

Unicode messages can be used if you need to send characters not available within an ordinary [text message](#) (e.g. Arabic, Japanese). A Unicode SMS can contain maximum 70 characters per message (or 67 characters per part for a long SMS).

Unicode messages are sent by setting parameter `type` to `unicode`. When sending Unicode messages, the parameter `charset` can be set to `UTF-8` to send data in the UTF-8 character set which holds characters of most of the world's languages.

Example to send text (Greece in Greek):

```
https://se-1.cellsynt.net/sms.php?
username=demo&password=test123&destination=00447920110000&type=unicode&charset=U
TF-8&text=%CE%95%CE%BB%CE%BB%CE%AC%CE%B4%CE%B1
```

## 3.3 Return value

In response to an HTTP request our SMS gateway gives a return value in plain text. There are two types of return values:

### 3.3.1 Successful request

OK: <space><trackingid>

A return value starting with OK: means the HTTP request and all its parameters were properly formatted and the message has been accepted for delivery. On the contrary it says nothing about the delivery of the message since that is not yet known when a message has been submitted. To get delivery status [Delivery reports](#) must be used.

In the return value is also a unique reference for the message that was sent (*trackingid*). This value is a string consisting of up to 40 characters.

Example:

```
OK: 92dff27302b754424242fb204620dc18
```

When sending long SMS you will get in return as many *trackingid* values comma separated as the number of parts in the message sent.

Example return value for a message which has been split into three parts (word-wrapped in this document but not in reality):

```
OK: de8c4a032fb45ae65ab9e349a8dc2458,
ed6037d0fe08dd4a4ab5cdcfd5aae653,6a351ae2ef03c3c5e271adcccd140089
```

### 3.3.2 Failed request

Error: <space><error message>

This return value is returned when erroneous parameters have been supplied in the HTTP request. This means that an application never should be constructed to under any circumstances retry



## Integrating with Cellsynt's SMS gateway via HTTP interface (technical documentation)

sending a message which have been given Error: in return as it always indicates a permanent error.

The error message returned is in plain text and indicates which parameter(s) that were missing or had faulty values.

Example:

```
Error: Parameter destination must be set
```

### 3.3.2.1 Out of credit

If your account's credit limit has been reached the following message will be returned:

```
Error: Out of credits
```

## 4 Delivery reports

Delivery reports enable you to receive information about a message's delivery status and is activated by specifying a URL in our web interface to which we then will start performing HTTP GET requests with the following parameters:

Parameter	Description
trackingid	The unique message reference returned when sending a message.
status	Current status in the operator network. For available values see table below.
destination	Recipient to which the message was sent.

Possible status values:

Status	Category	Description
buffered	Intermediate status	The operator has not yet been able to deliver the message due to problems in the operator network or in the mobile phone (inbox full etc). Attempts are made trying to deliver the message.
delivered	Final status	Message delivery was successful.
failed	Final status	Message delivery failed.

There is no guarantee that delivery reports will be received from operators for all sent messages. Temporary traffic peaks or technical issues in mobile networks can cause delivery reports to be delayed. All information we receive from operators will however be relayed to you immediately.

You should never construct application logic to rely on delivery reports (for example re-send messages when you receive a "failed" report). Delivery reports should only be seen as an indication of a message's current delivery status.

