

Serwer **SMS**.pl[®]
System Obsługi Marketingu Mobilnego

SOAP Documentation



Introduction	2
General messages	3
Sending SMS messages with the same content	7
Send personalized SMS messages	12
Example of integration in C #	15

Introduction

The SOAP service consists in configuring the HTTP connection in which the data structure is defined in XML format. This message contains the following elements (tags):

- envelope, contains message content,
- header, the optional element, contains header information,
- body, contains query and answer.

Currently, this service is no longer supported by its creators, so we recommend that our Customers use other interfaces available to connect to our Platform.

General messages

If you do not specify an action, blocked customer account or provide incorrect customer data, SerwerSMS.pl will generate a document in JSON or XML format with information about the error that occurred. For example, if you provide an incorrect login or password, the customer will see the following message:

```
{
  "error":
  {
    "code":1001,
    "type":"InvalidUser",
    "message":"Incorrect login or password"
  }
}
```

If no action is specified, the system will display the message:

```
{
  "error":
  {
    "code":1000,
    "type":"InvalidAction",
    "message":"No action has been specified"
  }
}
```

Preview the error message can be checked with the action:

error/[code]

example:

<https://api2.serwersms.pl/error/1000>

response:

```
{
  "error":
  {
    "code":1000,
    "type":"InvalidAction",
    "message":"No action has been specified"
  }
}
```

List of error codes returned by the AP

Below is a table of codes and associated text descriptions.

Code	Type	Description
General		
1000	InvalidAction	No action has been specified
1001	InvalidUser	Incorrect login or password

1002	InvalidRole	You do not have permission to use the API
1003	InvalidIP	Unauthorized IP address
1004	InvalidID	Invalid ID parameter
1005	InvalidRole	No permissions
1006	InvalidData	Incorrect data format
1007	InvalidRole	Account is not fully active
1008	InvalidRole	You do not have permission to view numbers
2000	ValidationRequiredError	No parameters required
2001	ValidationPhoneError	Invalid phone number
2002	ValidationCodeError	Invalid error code
Messages		
3000	SendError	The SMS ECO+ SMS limit has been exhausted
3001	SendError	Brak uprawnień do wysyłania wiadomości SMS ECO+
3002	SendError	No permission to send SMS ECO+
3003	SendError	No permissions to send FULL SMS
3004	SendError	The limit of available MMS messages has been exhausted
3005	SendError	No permission to send MMS
3006	SendError	The available VOICE message limit has been exhausted
3007	SendError	No permission to send VOICE messages
3008	SendError	No permission to send Flash SMS messages
3009	SendError	No permissions to send SMS
3100	SendError	Sending SMS messages has been blocked
3101	SendError	Message is empty
3102	SendError	Message exceeded the allowed number of characters
3103	SendError	Wrong recipient number
3104	SendError	No phone numbers listed
3105	SendError	Incorrect message content
3106	SendError	This sender name can not be used
3107	SendError	You do not have permission to send personalized messages
3108	SendError	Invalid dispatch time
3109	SendError	Invalid file

3110	SendError	Too many messages to send, the maximum is 100,000 SMS in one application
3111	SendError	Unique error
3200	SendError	Incorrect characters in unique_id
3201	SendError	Uploaded unique_id are not unique
3202	SendError	Inconsistent amount of personalized and unique_id
3330	MessageError	No message
Pliki		
4000	FileError	File is too large
4001	FileError	File does not exist
4002	FileInvalid	Invalid file
Phone number / HLR		
4100	PhoneError	Connection error
4101	PhoneError	Invalid number
4102	PhoneError	No permission to HLR
4103	PhoneError	HLR query limit has been exhausted
4104	PhoneError	Account lockout
4105	PhoneError	Waiting for the answer
4106	PhoneError	Invalid identifier
Premium		
4200	PremiumError	Dispatch error
4201	PremiumError	Incorrect data
Account		
4300	AccountError	No login
4301	AccountError	No password
4302	AccountError	No phone number
4303	AccountError	No email address
4304	AccountError	No first name
4305	AccountError	No last name
4306	AccountError	No company name
4307	AccountError	Invalid number
4308	AccountError	Incorrect e-mail address
4309	AccountError	Too short login

4310	AccountError	Invalid login
4311	AccountError	Incorrect company
4312	AccountError	Login exists
4313	AccountError	Registration error
4320	AccountError	No current password entered
4321	AccountError	Current password is invalid
4322	AccountError	No new password entered
4323	AccountError	New password entries are not identical
4324	AccountError	No login
4325	AccountError	Invalid login
Sender's name		
4400	SenderExists	The name of the sender already exists
4401	SenderError	It is not possible to add a number as the sender's name
4402	SenderInvalid	Invalid sender name
4403	SenderError	Invalid number
4404	SenderError	Invalid code or number
4405	SenderError	The number was already active
Subaccounts		
4500	SubaccountError	No permissions to users department
4501	SubaccountError	Subaccount can not create subaccount
4502	SubaccountError	Login exists
4503	SubaccountError	No permission template
4504	SubaccountError	Incorrect login or password
Black list		
4600	BlacklistError	No operations to be performed
4601	BlacklistPhoneExists	The number already exists in the list
4602	BlacklistError	No number was found

Sending SMS messages with the same content

Call address

To send an SMS message using the Remote Service, use the following code and send the appropriate parameters.

```
{  
  
require_once('lib/nusoap.php');  
  
try {  
  
    $wsdl_url = 'https://soap.serwersms.pl/server.php?wsdl';  
  
    $client = new nusoap_client($wsdl_url, 'wsdl');  
    $client->setCredentials("demo", "demo", "basic");  
    $result = $client->call('Messages.sendSms',  
        array( ' ', 'test', 'test', array( 'test' => 1, 'details' => 1) ));  
  
    var_dump($client->responseData,$result,$client->getError());  
    exit;  
  
} catch(Exception $e) { var_dump($e);exit; } }  
}
```

WARNING! To send an SMS FULL message fill in the sender field (alphanumeric name or number). If the sender parameter is omitted or its value is empty, an ECO type SMS will be sent. Filling the parameter of the sender determines whether ECO or FULL will be sent.

Available parameters

Parameter	Type	An example value or format	Description
username	String	login	Username used to dispatch the message
password	String	password	Password to the account
phone	String Array	+48500600700	Number or table phones numbers .
text	String	content	Message content.
flash	Boolean	true, false or none	An optional parameter.
test	Boolean	true, false or none	The optional parameter, so you can check out the question of sending SMS messages. In response document is generated JSON / XML identical to the standard question, but the message will not be sent. Useful for testing applications (debug).

sender	String	The alphanumeric name of the sender for example. "INFORMATION" own name or number 4 or 9-digit.	The optional parameter allows you to change the sender field of SMS messages to almost any name or number. Each name of the sender must first be added to the Panel or API, and then approved by the administrator SerwerSMS.pl.
wap_push	String	Address URL or none	An optional parameter.
utf	Boolean	true, false or none	An optional parameter allows you to send SMS FULL containing eg. Polish diacritics, Cyrillic, etc. Only available in SMS FULL. Using this option changes the maximum length of a single SMS from 160 to 70 characters.
details	Boolean	true, false or none	The parameter displays in response to feedback details sent messages.
vcard	Boolean	true, false or none	An optional parameter, which allows sending messages VCARD.
speed	Boolean	1, 0 or none	An optional parameter allows you to send an independent channel with the highest quality (extra charge).
date	DateTime	ISO eg. „2015-02-22 12:25:55”	An optional parameter allows you to specify the date of dispatch messages
unique_id	String Array	eg. 6asTD3fif98gj	An optional parameter allows you to define your own ID sent message. The identifier can be a minimum of 3 characters and a maximum of 50 alphanumeric characters (a-z, A-Z, 0-9). For group mailings, another unique_id must be unique and the number unique_id must be consistent with the amount of numbers.
group_id	String Array	eg. 123456789	ID or group IDs in the Customer Panel. These identifiers can be downloaded using action groups / index or by copying them from the editing group in the Customer Panel.
contact_id	String Array	eg. 123456789	ID or IDs of contacts in the Customer Panel. These identifiers can be downloaded using the stock contacts / index.

dlr_url	String	http://www.twojadres.pl/skrypt.php?smsid=#SMSID#&stan=#STAN#&data=#DATA#	<p>This parameter allows the transmission method Push delivery reports on URL the Client.</p> <p>Setting this option to query the API overrides that in the settings in the Customer Panel. The address should be passed in encrypted form by the urlencode (). Address to forward reports applies to all messages submitted for execution within a single query.</p> <p>Available parameters:</p> <p>#SMSID# - message ID</p> <p>#STAN# - condition of delivery</p> <p>#DATA# - date of status change</p> <p>#PRZYCZYNA# - the reason for non-delivery messages</p>
---------	--------	--	--

The parameters in bold are obligatory. Others are optional.

If you wish to send a message as a vCard (parameter "vcard = true"), the structure of the message must be in a particular format. An example of the Vcard format can be found in the generally available documentation on the Internet or using the Customer Panel and SMS form. The next tags in the Vcard message should be separated by a newline character.

The FULL SMS message that contains Polish special characters should be properly encoded in UTF-8 before sending. To send a message with Polish characters, the "utf = true" parameter must be set, the subscriber must be authorized to send SMS messages FULL and the sender field must be filled in (this means that an SMS FULL will be sent).

Return response

Depending on the data sent, SerwerSMS.pl will generate in the appropriate document in the JSON / XML format with information about the actions carried out. In the case of correctly sending an SMS message, the customer will receive, for example, the following information:

```
{
  "success":true,
  "queued":1,
  "unsent":0
}
```

If you provide an additional parameter details = true, the reply will be supplemented with the details of sent messages that can be saved in the database on the client software side:

```
{
  "success":true,
  "queued":1,
  "unsent":0,
  "items":[{
    "id":"1c142d81c7",
    "phone":"+48500600700",
    "status":"queued",
    "queued":"2014-10-16 16:49:05",
    "parts":1,
    "text":"Test SerwerSMS.pl"
  }]
}
```

The "success" parameter contains information about the success of the operation. The "queued" and "unset" attributes include the number of queued and unsent messages. The "items" section contains telephone numbers and ID of messages to be sent (and messages that have not been queued for a specific reason). The unique SMS tag can be used later to check the

status of sending a specific SMS message remotely. The "text" parameter shows the content of the sent SMS message. The telephone number is automatically corrected and displayed in the full format required by SerwerSMS.pl, ie with the country code (eg +48) at the beginning.

In the case of providing, for example, a wrong phone number, an error message will be generated as well as a message ID (so that you can check later when and why it was not sent). For example, if two correct numbers and two wrong numbers are given, the system will generate the following JSON document:

```
{
  "success":true,
  "queued":2,
  "unsent":2,
  "items":[{
    "id":"32039da8e9",
    "phone":"+48500600700",
    "status":"queued",
    "queued":"2014-10-20 12:32:52",
    "parts":1,
    "text":"Test platform SerwerSMS.pl"
  },
  {
    "id":"844c2b4af0",
    "phone":"+48500600500",
    "status":"queued",
    "queued":"2014-10-20 12:32:52",
    "parts":1,
    "text":"Test platformy SerwerSMS.pl"
  },
  {
    "id":"6e373d7856",
    "phone":"+48500600500",
    "status":"unsent",
    "queued":"2014-10-20 12:32:52",
    "error_code":3103,
    "error_message":"Incorrect number",
    "text":"Test platform SerwerSMS.pl"
  },
  {
    "id":"bf069fe2c1",
    "phone":"+48500600500",
    "status":"unsent",
    "queued":"2014-10-20 12:32:52",
    "error_code":3103,
    "error_message":"Incorrect number",
    "text":"Test platform SerwerSMS.pl"
  }
]
```

In addition, a general error can be generated where there is no delimitation to the queued and incorrect. This can happen, for example, when the customer does not define the message content, does not provide phone numbers, his account is not active or there is another problem described in the error messages. If you do not enter the message, the following message will be generated:

```
{
```

```
"error":{
  "code":3101,
  "type":"SendError",
  "message":"The message is empty"
}
}
```

and similarly if no phone numbers are provided:

```
{
  "error":{
    "code":3104,
    "type":"SendError",
    "message":"No specified phone numbers"
  }
}
```

Recommended settings

In the case of medium and large amounts of sent messages of a few thousand or more, it is recommended to forward messages in "packs" of about 50-200 numbers in one question. This will greatly speed up the process of data transfer to SerwerSMS.pl and reduce the number of queries necessary to send.

If you try to send messages to two or more of the same phone numbers in a single query, the system will only send a message once to that number.

Send personalized SMS messages

Calling the address

For using remote control to send personalized SMS message, call a specific URL POST method. In the messages personalized there are no parameters like "phone" or "text". Instead, there is an additional parameter "messages", which contains a set of numbers assigned to them and the individual message. The value of "messages" consists of a two-dimensional array, which contains the keys "phone" and "text". Limit a single call to this question is 10 000 numbers and corresponding content available messages (recommended are much smaller chunks of data, approx. 50-200 numbers in a single query). Examples of calling dispatch may look like this:

```
{  
  
require_once('lib/nusoap.php');  
  
try {  
  
    $wsdl_url = 'https://soap.serwersms.pl/server.php?wsdl';  
  
    $client = new nusoap_client($wsdl_url, 'wsdl');  
    $client->setCredentials("demo", "demo", "basic");  
    $result = $client->call('Messages.sendPersonalized',  
        array( array(  
            array('phone' => '5000000001','text'=>'test3'),  
            array('phone' => '5000000002','text'=>'test4') ), 'sobek',  
            array( 'test' => 1, 'details' => 1, 'group_id' => '17324179' ) ));  
  
    var_dump($client->responseData,$result,$client->getError());  
    exit;  
  
    } catch(Exception $e) { var_dump($e); exit; } }  
}
```

Calling the above action with the given parameters (in the case of correct data to log) would send an SMS with the text "message1" to the number 500 600 700 and the second SMS message "message2" to the number 500 600 500.

Available parameters

Parametr	Type	An example value or format	Description
username	String	login	Username used to dispatch the message
password	String	password	Password to the account
messages	Array	[array('phone' => numer, 'text' => message), array('phone' => numer, 'text' => message)]	Phone numbers assigned to them the content of the SMS.
details	Boolean	true, false or none	The parameter displays in response to feedback details sent messages.
flash	Boolean	true, false or none	An optional parameter.

test	Boolean	true, false or none	The optional parameter, so you can check out the question of sending SMS messages. In response document is generated JSON / XML identical to the standard question, but the message will not be sent. Useful for testing applications (debug).
sender	String	The number in the format +48500600700 or proper name (max 11 characters, uppercase and lowercase letters and numbers, and a space character).	An optional parameter allows you to change the sender field of SMS messages to any number or name. You can only use the name approved by administrators or pre-defined names
utf	Boolean	true, false or none	An optional parameter allows you to send SMS messages FULL containing, among others, Polish diacritics.
voice	Boolean	true, false or none	An optional parameter allows you to send SMS messages VOICE. Parameter sender must remain empty.
date	DateTime	ISO eg.: „2010-11-09 15:23”	An optional parameter allows you to specify the date of dispatch of SMS messages.
unique_id	String Array	eg. 6asTD3fif98gj	An optional parameter allows you to define your own ID message you send. The identifier can be a minimum of 3 characters and a maximum of 50 alphanumeric characters (a-z, A-Z, 0-9). For group mailings, another unique_id must be unique, given in the form of a table and the number of unique_id must be consistent with the amount of personalized messages.

The parameters in bold are obligatory. Others are optional.

Return generated after the message is the same as the regular sending of SMS messages.

```
{
  "success":true,
  "queued":2,
  "unsent":2,
  "items":[{"
    "id":"98c93cc5f2",
    "phone":"+48500600700",
    "status":"queued",
    "queued":"2014-10-20 13:26:50",
    "parts":1,
    "text":"message1"
  }],
  {
    "id":"a20ff834c8",
```

```

    "phone":"+48500600500",
    "status":"queued",
    "queued":"2014-10-20 13:26:50",
    "parts":1,"text":"message2"
  },
  {
    "id":"b25462142e",
    "phone":"45648",
    "status":"unsent",
    "queued":"2014-10-20 13:26:50",
    "error_code":3103,
    "error_message":"Incorrect number",
    "text":"message3"
  },
  {
    "id":"d862cc6575",
    "phone":"11999885544",
    "status":"unsent",
    "queued":"2014-10-20 13:26:50",
    "error_code":3103,
    "error_message":"Incorrect number",
    "text":"message4"
  }
}

```

Recommended settings

In the case of medium and large amounts of messages sent several thousand or more, it is recommended to transfer messages in "packages" after approximately 50-200 numbers in a single query. This will speed up considerably the process of transferring the data to SerwerSMS.pl and reduce the amount necessary to send queries. Define the parameter "messages" overwrites parameter values `phone` and `text`.

Example of integration in C

To be able to use the SOAP service, you must add Service References in your project specifying the url as a parameter:
<https://soap.serwersms.pl/server.php?wsdl>

The following is an example of how to send an SMS via this service:

```
{

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Web;
using System.ServiceModel;
using System.ServiceModel.Channels;

var service = new namespace.Service.SerwerSMSPlPortTypeClient();
service.ClientCredentials.UserName.UserName= "demo";
service.ClientCredentials.UserName.Password = "demo";

try {

String encoded = System.Convert.ToBase64String(System.Text.Encoding.GetEncoding("UTF-8")
    .GetBytes(service.ClientCredentials.UserName.UserName + ":" + service.ClientCredentials.UserName.Password));

HttpRequestMessageProperty request = new HttpRequestMessageProperty();
request.Headers[System.Net.HttpRequestHeader.Authorization] = "Basic " + encoded;
OperationContextScope clientScope = new OperationContextScope(service.InnerChannel);
OperationContext.Current.OutgoingMessageProperties.Add(HttpRequestMessageProperty.Name, request);

soap.Service.paramsMessageSendSms param = new namespace.Service.paramsMessageSendSms { };
param.details = true;
var response = service.messages_sendSms("500000002", "sender", "text", param);
Console.WriteLine(response);

} catch(Exception e) { Console.WriteLine("{0} Exception caught.", e); } }

}
```