

# Monema Client SOAP API

1. SOAP API WSDL.....	3
1.1 WSDL LOCATION.....	3
1.2 PHP 5 SAMPLE: Get the current credit of an account.....	3
1.3 PHP 5 SAMPLE: Send a Fax.....	4
2. COMPLEX TYPES.....	5
2.1. Calls.....	5
2.2. CallsTags.....	5
2.3. Extensions.....	5
3. OPERATIONS.....	7
3.1. C_getCredit.....	7
3.2. C_getCalls.....	7
3.3. C_getCallTags.....	7
3.4. C_makeCall.....	8
3.5. C_makeCallResponse.....	8
3.6. C_getCallDownloadLink.....	9
3.7. C_getExtensions.....	10
3.8. C_enableExtensionTransfer.....	10
3.9. C_disableExtensionTransfer.....	10
3.10. C_getExtensionTransferStatus.....	11
3.11. C_sendSMS.....	11
3.12. C_getFaxModems.....	12
3.13. C_startFaxUpload.....	12

3.14. C_uploadFaxChunk.....	13
3.15. C_sendFax.....	13

# 1. SOAP API WSDL

## 1.1 WSDL LOCATION

Monema Client SOAP API WSDL is located in each server in this location:

`https://<IP>/public_api/pb_tamtam_wsdl.php`

Where <IP> is the IP or hostname of the server associated with your account.

## 1.2 PHP 5 SAMPLE: Get the current credit of an account

```
<?php
echo "Monema SOAP API sample\n";
echo "Creating soap client object...";
$SERVER_IP = 'YOUR ACCOUNT SERVER IP';
$ACCOUNT_LOGIN = 'YOUR ACCOUNT LOGIN';
$ACCOUNT_PASSW = 'YOUR ACCOUNT PASSWORD';
try
{
    $sc = new SoapClient('https://'.$SERVER_IP.'/public_api/pb_tamtam_wsdl.php');
    echo "Getting account credit... \n";
    $credit = $sc->C_getCredit($ACCOUNT_LOGIN , $ACCOUNT_PASSW);
    echo "Credit: $credit\n";
}
catch(Exception $e)
{
    echo "Error invoking API: ".$e->getMessage()."\n";
}
?>
```

## 1.3 PHP 5 SAMPLE: Send a Fax

```
<?php
echo "Monema SOAP API sample to send a fax\n";
echo "Creating soap client object...";
$SERVER_IP = 'YOUR ACCOUNT SERVER IP';
$ACCOUNT_LOGIN = 'YOUR ACCOUNT LOGIN';
$ACCOUNT_PASSW = 'YOUR ACCOUNT PASSWORD';
try
{
    $sc = new SoapClient('https://'.$SERVER_IP.'/public_api/pb_tamtam_wsd1.php');
    echo "Getting list of modems... \n";
    $modems = $soap_client->C_getFaxModems($ACCOUNT_LOGIN, $ACCOUNT_PASSW);
    if(!empty($modems))
    {
        echo "Starting fax upload...\n"
        $session_id = $soap_client->C_startFaxUpload($ACCOUNT_LOGIN,
$ACCOUNT_PASSW);
        $chunk_size = 256*1024;
        $file_to_upload = '/tmp/your_pdf_document.pdf';
        $file_size = filesize($file_to_upload);
        $f = fopen($file_to_upload,"r");
        while(!feof($f))
        {
            $buf = fread($f, $chunk_size);
            $size = $soap_client->C_uploadFaxChunk($ACCOUNT_LOGIN, $ACCOUNT_PASSW,
$session_id, $buf);
        }
        fclose($f);

        echo "Sending fax...\n";
        $ok = $soap_client->C_sendFax($ACCOUNT_LOGIN, $ACCOUNT_PASSW,$session_id,
$modems[0], '1555000000',1,'DestName', 'DestCompany', 'DestLocation', 'DestVoicePhone', '
DestSubject', 'DestComments');
    }
    else
    {
        echo "Not modems found to send faxes\n";
    }
}
catch(Exception $e)
{
    echo "Error invoking API: ".$e->getMessage()."\n";
}
?>
```

## 2. COMPLEX TYPES

### 2.1. Calls

Element Name	Type	Description
id	integer	Unique ID of the call in the server
clients_id	integer	Identifier of client the calls belongs to
src	string	Phone number that originated the call
dst	string	Phone number that received the call
duration	integer	Duration of the call in seconds
disposition	string	Status of the call (ANSWERED/NO ANSWER/BUSY/CANCEL)
uniqueid	string	Unique ID of the call in the PBX
recorded	boolean	True if the call was recorded
calldate	integer	UNIX timestamp when the call was made
accountcode	integer	Internal use for billing purposes
cost	double	Cost of the call in the account default currency
ext_src_recorded	boolean	True if the caller recorded the call
ext_dst_recorded	boolean	True if the calle recorded the call

### 2.2. CallsTags

Element Name	Type	Description
id	integer	Unique ID of the call tag in the server
calls_id	integer	Identifier of the call the tag belongs to
clients_id	integer	Identifier of client the tag belongs to
tag	string	Text of the tag
uniqueid	string	Unique ID of the call in the PBX

### 2.3. Extensions

Element Name	Type	Description
id	integer	Unique id of the extension in the PBX
users_id	integer	Unique id of the user associated to the extension in the server
clients_id	integer	Identifier of the client the extension belongs to
extnumber	integer	Number of the extension

### 3. OPERATIONS

#### 3.1. C\_getCredit

This operation allows to get the current credit balance of an account.

##### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account

##### *Returns*

Element Name	Type	Description
credit	double	Current credit balance of the account

#### 3.2. C\_getCalls

Get the list of calls made or received in the PBX phone system.

##### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
ClientPassword	string	Password of the account
dateStart	string	Return calls later to this date
dateEnd	string	Return calls previous to this date
fromId	string	Return calls with an ID greater than value provided
type	string	Filter calls by any of this types:  “outgoing”: returns only calls made to external numbers. “incoming”: returns only calls received in the PBX from external numbers. “internal”: returns only calls made between extensions

##### *Returns*

Element Name	Type	Description
calls	Calls[]	Array of Calls complex type with the list of selected calls

#### 3.3. C\_getCallTags

Get the list of tags of a given call.

### Input Parameters

Element Name	Type	Description
clientLogin	string	Login of the account
ClientPassword	string	Password of the account
callId	string	Call's ID

### Returns

Element Name	Type	Description
calltags	CallsTags[]	Array of CallsTags complex type with the list of tags of the selected call.

## 3.4. C\_makeCall

Place a call between two extensions.

### Input Parameters

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
SourceExt	string	Number of the extension that originates the call. PBX will ring this extension and, when it answers, the call will be made to the destination extension
destExt	string	Number of the extension to be called. It is also possible to call an external number
variables	string	List of variables to define in the call. The variable names are the same used in the components in the control panel Dial Plan. Must be defined between "{" and "}" characters and separated from the value with the "=" character.  Multiple variables can be defined separated by " " character:  {VAR1}="VALUE1" {VAR2}="VALUE2" {VAR3}=...

### Returns

Element Name	Type	Description
ok	boolean	Number of the extension to be called

## 3.5. C\_makeCallResponse

Place a call between two extensions and waits until SourceExt answers the call. The status and uniqueId of the call is returned.

### Input Parameters

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
SourceExt	string	Number of the extension that originates the call. PBX will ring this extension and, when it answers, the call will be made to the destination extension
destExt	string	Number of the extension to be called. It is also possible to call an external number
variables	string	List of variables to define in the call. The variable names are the same used in the components in the control panel Dial Plan. Must be defined between "{" and "}" characters and separated from the value with the "=" character.  Multiple variables can be defined separated by " " character:  {VAR1}="VALUE1"  {VAR2}="VALUE2"  {VAR3}=...

#### Returns

Element Name	Type	Description
ok	boolean	Number of the extension to be called
status	string	NO_SUCH_EXTENSION => Extension not valid NO_ANSWER => SourceExt didn't answered the call ANSWERED => SourceExt answered the call BUSY => SourceExt is busy NOT_AVAILABLE => SourceExt is not available right now
uniqueid	string	Unique identifier of the call made

### 3.6. C\_getCallDownloadLink

Get an URL to download specified call

#### Input Parameters

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
callId	integer	The identifier of the call returned from C_getCalls (field id)

#### Returns

Element Name	Type	Description
url	string	URL to download the specified call



### 3.7. C\_getExtensions

Get the list of extensions created

#### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account

#### *Returns*

Element Name	Type	Description
extensions	Extensions[]	Array of Extensions complex type with information about each extension

### 3.8. C\_enableExtensionTransfer

Set a permanent transfer for an extension to another extension or an external number.

#### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
extensionId	integer	Identifier of the extension to modify their transfer status. This identifier is the "id" element returned for each extension with the C_getExtensions operation
dstNumber	string	Number of the extension or external number that will receive the calls
isExternal	boolean	If dstNumber is an external number, this value must be True

#### *Returns*

Element Name	Type	Description
ok	boolean	True if the transfer has been properly set

### 3.9. C\_disableExtensionTransfer

Disable all active transfers for an extension.

### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
extensionId	integer	Identifier of the extension to disable the transfer. This identifier is the "id" element returned for each extension with the C_getExtensions operation

### *Returns*

Element Name	Type	Description
ok	boolean	True if the transfer has been properly disabled

## **3.10. C\_getExtensionTransferStatus**

Get the transfer status for an extension, returning information about if it is enabled, the destination number and if the number is an extension or an external number.

### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
extensionId	integer	Identifier of the extension to get the transfer status. This identifier is the "id" element returned for each extension with the C_getExtensions operation

### *Returns*

Element Name	Type	Description
enabled	boolean	True if the transfer is enabled
number	string	Destination number where the calls are transferred
is_external	boolean	True if the destination number is an external number, or False if it is an extension

## **3.11. C\_sendSMS**

Send a SMS to a external number.

### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account

clientPassword	string	Password of the account
callerId	string	Number to be used as CallerID for the SMS
dstNumber	string	Destination number that will receive the SMS
message	string	Content of the SMS

*Returns*

Element Name	Type	Description
ok	boolean	True if the SMS has been properly sent

### 3.12. C\_getFaxModems

Get a list of fax modems assigned to the account.

*Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account

*Returns*

Element Name	Type	Description
modems	string[]	List of modems identifiers

### 3.13. C\_startFaxUpload

Star a fax session to upload the content of the file to be sent.

*Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account

*Returns*

Element Name	Type	Description
sessionId	string	Returns a value that identifies the upload session. This value is needed in the operations C_uploadFaxChunk and C_sendFax

### 3.14. C\_uploadFaxChunk

Upload a chunk of data of the content to be sent as fax.

#### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
sessionId	string	Unique identifier of the session started by C_startFaxUpload
chunk	base64Binary	Base64 encoded data of the content to be upload

#### *Returns*

Element Name	Type	Description
file_size	integer	Total size of the file uploaded at this time

### 3.15. C\_sendFax

Send a fax to one or several destination numbers.

#### *Input Parameters*

Element Name	Type	Description
clientLogin	string	Login of the account
clientPassword	string	Password of the account
sessionId	string	Unique identifier of the session started by C_startFaxUpload
faxModem	string	Identifier of the fax modem to be used to send the fax. This value is one of the returned by operation C_getFaxModems
destNumbers	string	Destination numbers that will receive the fax, separated by “,”
useCover	boolean	True if a cover should be created for the fax
coverDestName	string	Name of the destination person that will receive the fax
coverDestCompany	string	Name of the destination company
coverDestLocation	string	Location where the fax will be received
coverDestVoicePhone	string	Phone of the person/company that will receive the fax
coverDestSubject	string	Subject to be added to the cover
coverDestComments	string	Comments to be added to the cover

#### *Returns*

Element Name	Type	Description
--------------	------	-------------

ok	boolean	True if the fax has been properly sent
----	---------	--