GOLD MOBILE

Gold Mobile

GoMo Campaign Web Services v2.9.2

Documentation

CONFIDENTIAL
4/13/2016

Contents

1. Overview

This document provides documentation for the GoMo Campaign API. Incoming data is sent as URL encoded name/value pairs. A user can utilize the API using the login credentials specific to the account.

SSL is supported for all client connections. Replace HTTP with HTTPS in your connection URL. It should be noted that text messaging is not a secure platform. While we can encrypt data transmitted over the API there are many other steps in the SMS transmission that are not secure. Any data transmitted via SMS can be intercepted and should be treated as unsecure.

2. API Interface and Operations

The following are the API operations that are currently available. Each operation requires different parameters to be sent. Once a call to the API is made to the server, a response is sent to the client.

Note: When sending parameters, include 'action' and set it to the name of the API function.

2.1 Response Types

The API can return multiple response types depending on what your application prefers. To do this you need to send a custom HTTP header with the value of the response type you wish to use.

The custom header is X_API_TYPE. Acceptable values are:

- XML
- JSON

2.1.1getSession

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Obtains a sessionid (token) to be passed while calling API functions

Parameters

Name	Data Type	Use	Comments
action	string	Req	*getSession*
user	string	Req	API Username
password	string	Req	API Password

Response

Name	Data Type	Comments
sessionid	string	This is the token to be instantiated and used for each API call. Expires after 4 hours (renews each time it is used within the allotted 4-hour timespan).
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message

2.1.2sendBlast

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Sends a single Blast to all users in a category
Parameters

Name	Data Type	Use	Comments
action	string	Req	*sendBlast*
sessionid	string	Req	Token created using getSession
message	string	Req	160 character limit
title	string	Not Req	Title of message (subject for email)
type	string	Req	'*sms*', '*email*', or '*text*'
cid	array	Req	Category ID(s)
date	string	Req	Date and time to send Blast. Should be formatted as: *YYYY-MM-DD HH:MM:SS*

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message
id	integer	Blast ID number

2.1.3getBlastStatuses

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Get Blast result per individual SMS

Parameters

Name	Data Type	Use	Comments
action	string	Req	*getBlastStatuses*
sessionid	string	Req	Token created using getSession
mid	integer	Req	Message ID

Response

Name	Data Type	Comments
message	string	Status message
status	string	Status (OK or Error)
statuscode	integer	Status code
item	container	
messageStatus	string	(Started, Sent, Queued, Failed)
messageID	integer	Message ID
phone	integer	mobile number
email	string	

2.1.4sendSMS

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Sends a single SMS to a mobile phone
Parameters

Name	Data Type	Use	Comments
action	string	Req	*sendSMS*
sessionid	string	Req	Token created using getSession
mobile	integer	Req	10-digits, no special characters
message	string	Req	160 character limit
cid	array	Req	Category ID(s)

Response

Name	Data Type	Comments
id	integer	Message ID
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message

2.1.5sendEmail

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Sends a single email to an address

Parameters

Name	Data Type	Use	Comments
action	string	Req	*sendEmail*
sessionid	string	Req	Token created using getSession
email	integer	Req	email address
message	string	Req	
cid	array	Req	Category ID(s)

Response

Name	Data Type	Comments
id	integer	Message ID
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message

2.1.6getMessageStatus

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Get the sent status of a message

Parameters

Name	Data Type	Use	Comments
action	string	Req	*getMessageStatus*
sessionid	string	Req	Token created using getSession
mid	integer	Req	Message ID

Response

Name	Data Type	Comments
messageStatus	string	(Started, Sent, Queued, Failed)
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message
reason	string	Reason code

2.1.7addSubscriber

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Adds a user to the subscriber table and/or adds to a list of specified categories
Parameters

Name	Data Type	Use	Comments
action	string	Req	*addSubscriber*
sessionid	string	Req	Token created using getSession
first_name	string		First name
last_name	string		Last name
email	string	Req*	Email address
mobile_country_code	integer		Defaults to 1
mobile	integer	Req*	10-digits, no special characters
zip	integer		5-digits
dob			User's birthdate. Should be formatted as: *YYYY-MM-DD HH:MM:SS*
gender	string		*M* or *F*
comments	string		Optional comment field
comments_1	string		Optional comment field
custom_1	string		Optional comment field
cid	array	Req	Category ID(s) to subscribe to.
optin	array		Set as *1* or *0*. Specifies if user is “opted in” to receive messages for the respective category.

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message

2.1.8editSubscriber

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Edits a subscriber and/or adds to a list of specified categories
Parameters

Name	Data Type	Use	Comments
action	string	Req	*editSubscriber*
sessionid	string	Req	Token created using getSession
sid	integer	Req	Subscriber ID
first_name	string		First name
last_name	string		Last name
email	string	Req*	Email address
mobile_country_code	integer		Defaults to 1
mobile	integer	Req*	10-digits, no special characters
zip	integer		5-digits
dob			User's birthdate. Should be formatted as: *YYYY-MM-DD HH:MM:SS*
gender	string		*M* or *F*
comments	string		Optional comment field
comments_1	string		Optional comment field
custom_1	string		Optional comment field
cid	array	Req	Category ID(s) to subscribe to.
optin	array	Req	Set as *1* or *0*. Specifies if user is “opted in” to receive messages for the respective category.

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message

2.1.9viewSubscriber

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Returns all data of a subscriber
Parameters

Name	Data Type	Use	Comments
action	string	Req	*viewSubscriber*
sessionid	string	Req	Token created using getSession
sid	integer	Req	Subscriber ID

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message
first_name	string	First name
last_name	string	Last name
email	string	Email address
mobile_country_code	integer	
mobile	integer	10-digits, no special characters
zip	integer	5-digits
dob		User's birthdate. Should be formatted as: *YYYY-MM-DD HH:MM:SS*
gender	string	*M* or *F*
comments	string	Optional comment field
comments_1	string	Optional comment field
custom_1	string	Optional comment field
cid	array	Category ID(s) to subscribe to.
optin	array	Set as *1* or *0*. Specifies if user is “opted in” to receive messages for the respective category.

2.1.10 deleteSubscriber

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Removes subscriber from table
Parameters

Name	Data Type	Use	Comments
action	string	Req	*deleteSubscriber*
sessionid	string	Req	Token created using getSession
sid	integer	Req	Subscriber ID

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message

2.1.11 listSubscribers

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Lists all subscribers in a specified category

Parameters

Name	Data Type	Use	Comments
action	string	Req	*listSubscribers*
sessionid	string	Req	Token created using getSession
cid	integer	Req	Category ID
lowerLimit	integer	Opt	Records returned will begin at this number.
upperLimit	integer	Opt	Records returned will end at this number.
startDate	date	Opt	Records returned will start at this date. Should be formatted as: *YYYY-MM-DD HH:MM:SS*
endDate	date	Opt	Records returned will end at this date. Should be formatted as: *YYYY-MM-DD HH:MM:SS*

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message
total	integer	Total number of subscribers
item	array	List of subscribers within the specified category. Returns as: *<item>* *<id> </id>* *<first_name></first_name>* *<last_name></last_name>* *<mobile></mobile>* *<email></email>* *<optin_time> </optin_time>* *<zip></zip>* *<dob>0000-00-00 00:00:00</dob>* *<gender></gender>* *<comments></comments>* *<comments_1></comments_1>* *<custom_1></custom_1>* *</item>*

2.1.12 getUnsubs

URL http://*subdomain*.gomocampaign.com/api.php

Purpose List opted out subscribers by date range

Parameters

Name	Data Type	Use	Comments
action	string	Req	*getUnsubs*
sessionid	string	Req	Token created using getSession
lowerLimit	integer	Opt	Records returned will begin at this number.
upperLimit	integer	Opt	Number of records to retrieve
startDate	date	Opt	Records returned will start at this date. Should be formatted as: *YYYY-MM-DD HH:MM:SS*
endDate	date	Opt	Records returned will end at this date. Should be formatted as: *YYYY-MM-DD HH:MM:SS*

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message
total	integer	Total number of subscribers who opted out
id_*	integer	List of user IDs. Returns as: *<id_1>user_id</id_1>*
mobile_*	integer	Mobile phone number for user * Returns as: *<mobile_1>*5551234567*</mobile_1>*
email_*	string	Email address for user * Returns as: *<email_1>*sample@email.com*</email_1>*

2.1.13 getEbb

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Get a list of bounced emails

Parameters

Name	Data Type	Use	Comments
action	string	Req	*getEbb*
sessionid	string	Req	Token created using getSession
startDate	date	Opt	Records returned will start at this date. Should be formatted as: *YYYY-MM-DD HH:MM:SS*
endDate	date	Opt	Records returned will end at this date. Should be formatted as: *YYYY-MM-DD HH:MM:SS*

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message
total	integer	Total number of emails that bounced
id_*	integer	Bounce-back ID Returns as: *<id_1>*id*</id_1>*
email_*	string	Email address Returns as: *<email_1>*sample@email.com*</email_1>*
stamp_*	date/time	Date and time when bounce back occurred. Returns as: *<stamp_1>* YYYY-MM-DD HH:MM:SS *</stamp_1>*

2.1.14 addCategory

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Adds a category to the categories table
Parameters

Name	Data Type	Use	Comments
action	string	Req	*addCategory*
sessionid	string	Req	Token created using getSession
category_name	string		Name of the Category
server	string		Server
double_optin	string		0 or 1
confirm_msg	string		
thanks_msg	string		
keyword	string		Keyword
type	string	Req	Type of category sms, email
display name	string		
reply_to	string		*Optional replyto address, useful for spoofing. Note: no way to edit in UI yet*
delay	string		Number of minutes before message is sent.
attach_file	string		Full path of file to attach. Note: File needs to be on local host
attachment	string		Set to 1 to include attachment
email_lead	string		Set to 1 to ti email for each optin
email_lead_address	string		Email address used in above
reply_once	integer		0 or 1, will not send subsequent optin messages
age_gate	integer		Not fully implented
minimum_age	integer		Not fully implented

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
cid	integer	ID of the newly created category

2.1.15 editCategory

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Edits a category based on the id you provide
Parameters

Name	Data Type	Use	Comments
action	string	Req	*editCategory*
sessionid	string	Req	Token created using getSession
cid	integer	Req	Enter the ID of the category you plan to edit
category_name	string		Name of the Category
server	string		Server
double_optin	string		0 or 1
confirm_msg	string		
thanks_msg	string		
keyword	string		Keyword
type	string		Type of category sms, email
display_name	string		
delay	string		Optional comment field
attach_file	string		Optional comment field
attachment	string		Optional comment field
email_lead	string		Optional comment field
email_lead_address	string		
reply_once	integer		0 or 1
age_gate	integer		
minimum_age	integer		Minimum age

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code

2.1.16 viewCategory

URL http://*subdomain*.gomocampaign.com/api.php

Purpose View all data of a single a category based on the id you provide

Parameters

Name	Data Type	Use	Comments
action	string	Req	*viewCategory*
sessionid	string	Req	Token created using getSession
cid	integer	Req	Enter the ID of the category you want to see the data for.

Response

Name	Data Type	Comments
id	integer	ID of category
category_name	string	Name of the Category
server	string	Server
double_optin	string	0 or 1
confirm_msg	string	
thanks_msg	string	
keyword	string	Keyword
type	string	Type of category sms, email
display_name	string	
delay	string	Optional comment field
attach_file	string	Optional comment field
attachment	string	Optional comment field
email_lead	string	Optional comment field
email_lead_address	string	
reply_once	integer	0 or 1
age_gate	integer	
minimum_age	integer	Minimum age

2.1.17 deleteCategory

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Deletes a category based on the id you provide

Parameters

Name	Data Type	Use	Comments
action	string	Req	*deleteCategory*
sessionid	string	Req	Token created using getSession
cid	integer	Req	Enter the ID of the category you plan to edit

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code

2.1.18 listCategories

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Lists all categories
Parameters

Name	Data Type	Use	Comments
action	string	Req	*listCategories*
sessionid	string	Req	Token created using getSession
cid	integer	Req	Enter the ID of the category you plan to edit

Response

Name	Data Type	Comments
status	string	Status (OK or Error)
statuscode	integer	Status code
message	string	Status message
total	integer	Total number of categories
item	array	List of subscribers within the specified category. Returns as: *<item>* *<id> </id>* *<category_name></category_name>* *<shortcode></shortcode >* *</item>*

2.3.1checkEmailAvailability

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Check if email address is available

Parameters

Name	Data Type	Use	Comments
action	string	Req	*checkEmailAvailability*
sessionID	string	Req	Token created using getSession
email	string	req	
cid	integer	req	

Response

Name	Data Type	Comments
status	int	Success status value
statuscode	integer	Status code
message	string	Status message

2.3.2checkMobileAvailability

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Check if phone number is available
Parameters

Name	Data Type	Use	Comments
action	string	Req	*checkMobileAvailability*
sessionID	string	Req	Token created using getSession
mobile	integer	req	
cid	integer	req	

Response

Name	Data Type	Comments
status	int	Success status value
statuscode	integer	Status code
message	string	Status message

2.3.3getUserIDbyEmail

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Get user ID by email address

Parameters

Name	Data Type	Use	Comments
action	string	Req	*getUserIDbyEmail*
sessionID	string	Req	Token created using getSession
email	string	Req	Email Address

Response

Name	Data Type	Comments
status	int	Success status value
statuscode	integer	Status code
message	string	Status message
id	integer	user ID

2.3.4getUserIDfromMobile

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Get user ID from mobile number

Parameters

Name	Data Type	Use	Comments
action	string	Req	*getUserIDfromMobile*
sessionID	string	Req	Token created using getSession
mobile	integer	Req	mobile phone number

Response

Name	Data Type	Comments
status	int	Success status value
statuscode	integer	Status code
message	string	Status message
id	integer	All user ids with mobile, separated by commas

2.3.5isUnsubscribedEmail

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Check whether a subscriber is opted into an email category
Parameters

Name	Data Type	Use	Comments
action	string	Req	*isUnsubscribedEmail*
sessionID	string	Req	Token created using getSession
cid	integer	Req	Category ID
email	String	Req	

Response

Name	Data Type	Comments
status	int	Success status value
optin	integer	0 not opted in 1 opted in 2 not found 3 whitelisted 4 opted in and opted out
message	string	Status message

2.3.6isUnsubscribedMobile

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Check whether a subscriber is opted into a SMS category
Parameters

Name	Data Type	Use	Comments
action	string	Req	*isUnsubscribedEmail*
sessionID	string	Req	Token created using getSession
cid	integer	Req	Category ID
mobile	String	Req	

Response

Name	Data Type	Comments
status	int	Success status value
optin	integer	0 not opted in 1 opted in 2 not found 3 whitelisted 4 opted in and opted out
message	string	Status message

2.3.7unsubscribeFromAllByEmail

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Remove subscriber associated with this email address from all categories
Parameters

Name	Data Type	Use	Comments
action	string	Req	*unsubscribeFromAllByEmail*
sessionID	string	Req	Token created using getSession
email	String	Req	

Response

Name	Data Type	Comments
status	int	Success status value
affected_rows	integer	
message	string	Status message

2.3.8unsubscribeFromAllByMobile

URL http://*subdomain*.gomocampaign.com/api.php
Purpose Remove subscriber associated with this mobile number from all categories
Parameters

Name	Data Type	Use	Comments
action	string	Req	*unsubscribeFromAllByMobile*
sessionID	string	Req	Token created using getSession
mobile	String	Req	

Response

Name	Data Type	Comments
status	int	Success status value
affected_rows	integer	
message	string	Status message

2.3.9unsubscribeFromCID

URL http://*subdomain*.gomocampaign.com/api.php

Purpose Remove subscriber associated with this sid from the specified category

Parameters

Name	Data Type	Use	Comments
action	string	Req	*unsubscribeFromCID*
sessionID	string	Req	Token created using getSession
sid	integer	Req	Subscriber ID
cid	integer	Req	CID

Response

Name	Data Type	Comments
status	String	OK or Error
statuscode	integer	code
message	string	Status message

3. Two-Way Messaging – MO Push

This function allows the Gold Mobile platform to push messages without the need for constant polling. You specify the location of a listener URL and when an MO comes in we will do an HTTP post with a single string variable called “messagelist” that contains an XML blob with the message data.

MESSAGELIST VARIABLE:

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<ENVELOPE>
<MESSAGE-LIST>
<MESSAGE ID="#########" SRCID='' RELATEDTO=''>
<TYPE>IN</TYPE>
<FROM>PHONE</FROM>
<OPTIONS></OPTIONS>
<FROM>##########</FROM>
<TO>######</TO>
<SUBJECT></SUBJECT>
<BODY>AAAAAAAAAAA</BODY>
</MESSAGE>
</MESSAGE-LIST>
</ENVELOPE>
```

End Session

Input:

http://URL/endsession?sessionID=*zzzz*

Input Parameters:

sessionID – is the session id as returned by the most recent “Session Start” call that has not yet timed out. Please note sessionID is **CASE SENSITIVE**.

OUTPUT:

```
<?xml version="1.0" encoding="UTF-8"?>
   <envelope>
      <status>sssss</status>
   </envelope>
```

Output Parameters:

status – the status returned by the API. Anything other than “OK” means the message is rejected by the API. It does not mean that the message was sent, as there are other systems underneath the API that might reject the message in later processing.

NOTES:

This interface must be called if the service is to go dormant for a time period exceeding the session time out value.

4. Delivery Receipts

Our system has the ability to push delivery receipts for messages sent. These are the same receipts passed back from the carrier. It is important to know that not all carriers support delivery receipts and some only support negative acknowledgement methods (sending a receipt on failure only).

In order to receive delivery receipts you first need to record the message ID that gets returned to you when you send an MT. You then need to provide us with a URL designed to receive the receipt. It will come in as a POST with the variable called messagelist and it will be in the form of a small XML blob. The XML will contain the message ID and the delivery information. Here is the format of the XML blob:

```
<?xml version="1.0" encoding="UTF-8"?>
<envelope>
<messageid>####</messageid>
<STATUS>####</STATUS>
</envelope>
```

Our system also has the ability to be polled for delivery receipts. This is the preferred method if outbound traffic is high as inbound DRs can overload a system if not properly equipped. To retrieve delivery receipts you must first either pass or record a unique identifier for each message sent. Then periodically poll (at suggested 15 minute intervals) the fetchDR end point which will return a payload. Note you will get multiple receipts for each message those are:

1. Queued
2. Started
3. Delivered to carrier
4. Delivered

5. Failed

The end point to hit is:

https://apihostname/fetchdr?sessionid=SESSIONID&command=fetchdr

For some carriers it is possible for them to not report Delivered in which case you can assume delivery.

5. Scrubbing URL

The most critically important task in maintaining any “opt-in” list is proper scrubbing. Scrubbing is the task of removing phone numbers from your list of mobile devices that are no longer available or if the subscriber no longer wishes to receive messaging (i.e. they reply with the message STOP).

All API accounts MUST supply an opt-out or “scrubbing” URL. This URL will take an HTTP POST with a single string variable called *phonenumber*. This variable contains the phone number. The action that must occur when this URL receives this variable is to scrub the number supplied in the POST.

Our system will track what campaign’s a subscriber belongs to so no action is required on your part.

Your script must return one two values:

+ successfully processed

– unsuccessful

You may keep the record for historical reference but you must NOT send any additional messages (other than an opt-out confirmation) to the supplied mobile number. Of course, if the subscriber chooses to opt-in again you may resume sending messages but this trail must be auditable in the event of a dispute.

Failure to properly scrub your database can result in an immediate suspension and possible revocation of service.

6. Status Codes

Every API call returns a status code. Below are the codes with their respective descriptions.

Code	Description
0	Success
1	Invalid mobile number
2	Invalid title
3	Invalid message
4	Invalid type
5	Invalid username/password
6	Invalid category ID
7	Invalid Session ID or session timed out
8	Invalid date for blast
9	Invalid action
10	Database error
11	Invalid email
12	Invalid first name
13	Invalid last name
15	Error creating token
16	Invalid zip code
17	Invalid gender
18	Error adding subscriber to category
19	Error reading incoming payload
20	Error showing data from XML
21	Error removing subscriber from category
22	Warning: No changes were made while trying to edit subscriber
23	Error reading subscriber data from database
24	Error displaying token
25	Store has not been configured
26	No storefront users found with that mobile number
27	An error has occurred when trying to deposit to your account. Please double-check your billing information and try again
28	Invalid deposit amount
29	Invalid store ID
30	Store ID not found
31	Warning: No changes were made while trying to edit category.
32	Warning: No changes were made while trying to edit category. No ID provided.
33	Not Available
34	Invalid customer ID
35	Missing Parameter(s)
36	Error saving billing information
37	Error retrieving balance

7. Using cURL

To be able to send an API request, you will need to use cURL to post your data. Also, it is required that the output is parsed since it is returned as XML. Below is a sample of a function in PHP that will do this.

```
function getAPI($url, $payload=array()){
        $result='';
        $payload = http_build_query($payload);
        $ch = curl_init();
        $user_agent = "Mozilla/5.0 (compatible; MSIE 5.01; Windows NT 5.0)";
        curl_setopt($ch, CURLOPT_USERAGENT, $user_agent);
        curl_setopt($ch, CURLOPT_POST, true);
        curl_setopt($ch, CURLOPT_FOLLOWLOCATION,true);
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE);
        curl_setopt($ch, CURLOPT_URL, $url);
        curl_setopt($ch, CURLOPT_POSTFIELDS, $payload);
        $result = curl_exec($ch);
        $result = simplexml_load_string($result);

        return ($result !== false ) ? $result : 'Error decoding response.';
}
```

8. PHP Code Samples

In your code, you will need to send the proper credentials to the cURL function. You may use the following PHP code samples to get the proper returns:

getSession Example:

```
$params = array('action' => 'getSession', 'user' => $user,
'password' => $password);
$getSession =
getAPI(‘http://subdomain.gomocampaign.com/api.php’, $params);
```

The above will send 3 variables to the function within an array. The function sends that information to the URL specified. The response is returned as an object. To retrieve the response from the API, you may use the following:

```
$getSession->sessionid;
$getSession->status;
$getSession->statuscode;
```

sendSMS Example:

```
$params = array('action' => 'sendSMS', 'sessionid' => $sessionid,
'mobile' => '5551231234', 'type' => 'sms', 'message' => 'This is a
test.', 'cid' => $cid);
$sendSMS =
getAPI('http://subdomain.gomocampaign.com/api.php', $params);
```

The function sends the information to the URL specified. The response is returned as an object. To retrieve the response from the API, you may use the following:

```
$sendSMS->id;
$sendSMS->status;
$sendSMS->statuscode;
```

sendBlast Example:

```
$params = array('action' => 'sendBlast', 'sessionid' => $sessionid,
'message' => 'This is a test message.', 'title' => $title, 'type' =>
$type, 'cid' => $cid, 'date' => 'YYYY-MM-DD HH:MM:SS');
$sendBlast =
getAPI('http://subdomain.gomocampaign.com/api.php ', $params);
```

The function sends the information to the URL specified. The response is returned as an object. To retrieve the response from the API, you may use the following:

```
$sendBlast->status;
$sendBlast->statuscode;
```

MO PUSH

```
<?php
//XML Example
$newXML = "<?xml version=\"1.0\" encoding=\"UTF-8\"?><envelope><message-list>
<message id=\'586027\' srcid=\'\' relatedto=\'\'><type>IN</type><options></options>
<from>ctpp%3A%2F%2Fericsson%3A53016%2F%3Fcarrier%3D31004%26address%3D4444444444</from>
<to>cn%3DDove%2Ccn%3DCustomers%2Ccn%3DCG1%2Ccn%3DDev%2Cdc%3Dgold-group%2Cdc%3Dcom</to>
<subject></subject><body>bstest 255</body></message></message-list></envelope>";
function receiveMessage($incomingMessage){
    //Strip Slashes from $incomingMessage string variable
    $messagelist_xml = simplexml_load_string(stripslashes($incomingMessage));
    $mlist = array();
    //Decode and get retrieve info from string
    //$from contains all of the incoming information where the message came from
    $mlist['from'] = urldecode($messagelist_xml->{'message-list'}->message->from);
    //$messagebody contains the message which was texted in
    $mlist['messagebody'] = urldecode($messagelist_xml->{'message-list'}->message->body);
    //breaks message into an array separated by a space.
    $mlist['keywordarray'] = explode(" ",$mlist['messagebody']);
    //Gets the first word in the message body, which would be the keyword they used to text in or as a response, whatever they responded with.
    $mlist['keyword'] = $mlist['keywordarray'][0];
    //get phone number from the $from variable
    $mlist['phonenumber'] = substr($mlist['from'],-10);
    //get carrier code from the $from variable
    $mlist['carriercode'] = strstr($mlist['from'], '=');
    $mlist['carriercode'] = substr($mlist['carriercode'],1,5);
    return $mlist;
}

$messageinfo = receiveMessage($newXML);

echo "<pre>";
print_r($messageinfo);
echo "</pre>";
?>
```

9. Change Control

Date	Version	Author	Reason
12/16/10	1.5	Efren J. Belizario	Modified original document which was written before change control. Added change control, subscriber-related API calls, and formatted the document for better readability.
1/13/11	1.6	EJB	Included ‘title’ parameter for sendBlast.
1/14/11	1.7	EJB	Included getUnsubs (unsubscribers) and getEbb (email bounce-backs).
3/3/11	1.8	EJB	Included limits (date and #) for getUnsubs and getEbb actions.
8/30/11	1.9	EJB	addSubscriber parameter “optin” is now optional.
9/27/11	2.0	EJB	Added Store functions (2.12-2.15). Improved output of listSubscribers function (nested results).
11/29/11	2.1	EJB	Added limits/dates options to listSubscribers. Added first and last names to output of listSubscribers.
3/5/2012	2.21	MS	Established 2.2 as baseline version
4/3/2012	2.31	JS	StoreFront and GomoWallet API capabilities
5/4/2012	2.41	JS	New Storefront API
6/12/2012	2.45	MS	Added additional web services
8/16/2012	2.47	JC	Removed four API calls. Improved document.
9/26/2012	2.48	JC	Added limit and minimum to getSubProductsByParentID
10/01/2012	2.49	JC	Added searchByNameDesc() and searchByZipCode()
10/10/2012	2.5	JC	Added location_id parameter to searchByZipCode(). Updated table of contents.
10/16/2012	2.5.1	JC	Added store address to addStore() and getStore(). Updated table of contents. Updated getSubProductsByParentID() to includes groups and their associated items.
10/18/2012	2.5.2	JC	Fixed NAN bug in searchByZipCode(). Output of getBlastStatuses(), getStoreMenu(), getCart(), getLocationByStoreID(), getOrderKeyword(), getProductGroupsByStoreID(), getProducts(), getProductsByTimeRange(), getSalesTax(), getSalesTaxByStoreID(), getStoreProductsByGroup(), getStore() , getStores(), getSubProductsByParentID(), getOrderHistory(), showStoreSpecials(), searchByNameDesc(), searchByZipCode() was changed to be more organized.
10/25/2012	2.5.3	JC	Fixed getCart() bug. Add key to response for items added to the cart.
10/26/2012	2.5.4	JC	Removed j_pass, employee_id, badge_id parameters from addUser(). Changed masked_cc_number to unmasked as we need to whole number to do wallet stuff. Added user_id to return values in addUser().
11/7/2012	2.5.6	JC	Removed wait_time, payment_method, payg parameters from addUser(). Added carrier table.

11/14/2012	2.5.7	JC	Added mobile_country_code parameters to addSubscriber(), editSubscriber(), viewSubscriber(), and listSubscribers()
12/13/2012	2.6	JC	Added JSON response type, added searchByLatLong(), added products order to getOrderHistory(), when calling addUser() if the user already exists an error message is returned, Added CVV parameter to placeOrder
1/22/2013	2.6.1	DA	Removed getUnsub
1/30/2013	2.6.2	DA	Removed voice functions; fixed typos
2/13/2013	2.6.3	JHM	Updated AddUser function
2/27/2013	2.6.4	JR	Updated parameters for resetPassword and updateCartQuantity
3/19/2013	2.6.5	JC	Added comments, fulfillment_method, and pickup_time parameters to placeOrder()
3/25/2013	2.6.6	JC	Fixed BillingInfo.save() and Consumer.getBalanceList() to use standard format naming convention
3/29/2013	2.7	JHM	Removed getSalesTax
4/2/2013	2.7.1	JR	Changed billinginfo.save to updateCreditCard
5/10/2013	2.8.0	JHM/JR	Updates these functions: getStoreOrders, addToCart, addUser, updateShippingAddress, updateUser, getResetPW
5/14/2013	2.8.1	JHM	Updated letterhead, added getUnsub.
6/17/2013	2.8.2	JR	Added isUnsubscribedEmail and isUnsubscribedMobile
7/16/13	2.8.3	JR	Added unsubscribeFromAllByEmail and unsubscribeFromAllByMobile
11/26/13	2.8.4	JHM	Added MO Push, Delivery Receipts, Scrubbing URL
9/22/14	2.8.5	JR	Updates to isUnsubscribedEmail, isUnsubscribedMobile and getUserIDfromMobile
1/16/15	2.8.6	JHM	Updated return field list for listSubscriber
10/29/15	2.9	JHM	Added new function unsubscribeFromCID
1/17/16	2.9.1	JHM	getMessageStatus has been updated to include Reason Code
4/13/2016	2.9.2	JHM	Updated parameters in getUnsubs