

# A Quick Guide to the PHP SMS REST Client

## Integrating the Text Marketer API with your code

Integration Guide

Version 1.2

July 2013

[www.textmarketer.co.uk](http://www.textmarketer.co.uk)

## Contents

|                               |   |
|-------------------------------|---|
| Document Change Log.....      | 3 |
| Introduction.....             | 4 |
| Getting Started.....          | 4 |
| A Basic Command.....          | 4 |
| Testing with the Sandbox..... | 5 |
| Functions Available.....      | 5 |

## Document Change Log

| Date             | Version | Detail                         |
|------------------|---------|--------------------------------|
| 15 July 2013     | 1.2     | Updates for client version 1.4 |
| 19 April 2012    | 1.1     | Updates for client version 1.2 |
| 09 February 2012 | 1.0     | First publication              |

## Introduction

The REST API is a powerful tool for programmatically controlling your Text Marketer account. It is the most feature-rich of all our APIs and provides the most detailed information about the success/failure of your requests.

This document describes the PHP REST client library, which makes it really easy to use the REST API in your PHP code by just including the provided library file and making the appropriate function calls.

For more advanced use of, and errors returned by, the REST API please see the document **The Complete Text Marketer RESTful SMS Services API** also available from [www.textmarketer.co.uk/developers/](http://www.textmarketer.co.uk/developers/).

## Getting Started

You will need a copy of the PHP library TMRestClient.php, available from [www.textmarketer.co.uk/developers/](http://www.textmarketer.co.uk/developers/) (go to the *Available Libraries* section of the REST API page).

You must 'include' the library in your PHP code:

```
include_once('TMRestClient.php');
```

The library file needs to be in your include path, or you can specify the absolute/relative path in the argument above.

## A Basic Command

The following is a complete example of using the REST client library. The PHP REST client may throw exceptions, which means that you need to place any function calls in a try-catch block, as follows:

```
<?php
    include_once('TMRestClient.php');
    $tmClient = new TMRestClient('myuser', 'mypass', 'production');
    try {
        $credits = $tmClient->getCredits();
        echo "I have $credits credits.";
    } catch (Exception $ex) {
        // handle a possible error – there may be more than one
        $errors = $tmClient->getLastErrors();
        foreach($errors as $errorcode => $errormsg)
            echo "Code $errorcode: $errormsg";
    }
?>
```

That's it! Obviously you would need to change the code to use your own API username/password. You can find your API username and password (which may be different to your web interface username/password) via the web interface:

<https://messagebox.textmarketer.co.uk> (Settings > API Settings)

If you don't have an account, you can set one up for free at [www.textmarketer.co.uk](http://www.textmarketer.co.uk).

## Testing with the Sandbox

A sandbox system is available to you for testing. The sandbox will not modify your account in any way, but will respond normally to your requests. In other words if you use the sandbox to get the number of credits on your account, it will provide the correct number. However if you use it to transfer credits between accounts, it will simulate a successful transfer (assuming you have enough credits on the account, and provide valid target account details, etc.) but it will not actually modify the number of credits on either account.

Similarly, the sandbox system will not actually send SMS messages and will not deduct credits for simulated sends.

To use the sandbox simply specify the sandbox when you create the client object:

```
$tmClient = new TMRestClient('myuser', 'mypass', 'sandbox');
```

NOTE: The values returned when using the Sandbox may differ slightly from the values returned when using the production system.

## Functions Available

When you download the PHP library (see *Getting Started* above) you will find automatically generated documentation included. This is the best place to see what functions are available and what arguments they take.

A simplified list of functions is given below.

- addGroup – create a new 'send group'
- addNumbersToGroupWithName – add new numbers to a 'send group'
- addNumbersToGroup – DEPRECATED, use addNumbersToGroupWithName() instead
- createSubAccount – creates a new account that is a sub-account of your main account
- getCredits – get the number of credits available in your account
- getDeliveryReport – get the contents of a delivery report
- getDeliveryReports – get a list of the available delivery reports
- getGroup – get the numbers in a group
- getGroups – get a list of the available 'send' and 'merge' groups
- getKeywordAvailability – get the availability of a keyword for use on our 88802 short code number
- getKeyword – DEPRECATED, use getKeywordAvailability() instead (booleans returned)
- getLastErrors – get the last errors returned from the last function call (to the API)

- isLoginValid – check the username/password credentials used are valid
- sendSMS – send an SMS to a given recipient
- deleteSMS – delete a previously scheduled SMS
- transferCreditsToAccount – transfer credits to a specified account number
- transferCreditsToUser – transfer credits to a specified account username