

# Promotions API

## Reference Guide

### Table of Contents

- [Calling the Promotions API](#)
- [Standard Parameters](#)
- [Standard Results](#)
- [PHP Example](#)
- [Listing Contest Fields](#)
- [Exporting Contest Entries](#)
- [Listing Active Contests](#)
- [Adding a Contest Entry](#)
- [Voting on a Contest Entry](#)
- [Voting on a Contest Entry with voting limits](#)
- [Counting an Entry's Votes](#)
- [Obtaining the Server's Time](#)
- [Testing Your API Connection](#)
- [Checking the Current API Version](#)

## Calling the Promotions API

The first method that you'll encounter will be *shoutlet\_time*, a simple parameterless method that returns as its results the current Unix epoch time on the Promotions server that requires no authentication. We prefer that you use *shoutlet\_time* at least for the initial authentication simply to help avoid possible synchronization issues. The current time is returned with all responses so subsequent calls can be based off of that.

There are four parameters and two return values that are common to all Promotions API methods. These four parameters are required in every API method call and are detailed in the standard parameter list below.

For security reasons, an SSL connection should be used for all API calls.

### Standard Parameters

#### **apikey (required string)**

The Promotions API key that you have generated for the application.

#### **timestamp (required integer)**

The current epoch time. This can be obtained via the *shoutlet\_time* method, from the timestamp returned by most methods or, if your server clock is accurate, via a local system call.

#### **sha\_hash (required string)**

This is a SHA-512 digest obtained by concatenating the following values in the order specified:

- the API key you passed as the first parameter
- the timestamp (as used for the second parameter)
- your API secret
- the name of the method you are calling

#### **mode (required string)**

The string 'json', 'xml', or 'csv' used to specify the format that you wish the results of the method to be returned in.

### Standard Results

#### **error (integer)**

This field is always present. If no error occurred, the value of this field will be 0, otherwise it will be set to 1 and the result will also include a message field describing the error.

#### **time (integer)**

This is the epoch time on the server at the time that the response was returned. It is included purely as convenience, since it can be used to eliminate most of the need for calls to *shoutlet\_time* for subsequent calls.

## PHP Example

```
define('API_KEY',    'Your API Key');
define('API_SECRET', 'Your API Secret');
define('API_URI',    'https://promotions.spredfast.com/shoutlet/interface/');

function call_promotionst_api($method, $param=array(), $mode='json', $time=false) {
    // If a timestamp is provided, use it, otherwise call the shoutlet_time endpoint
    // for a fresh timestamp to use. Note: timestamps are part of every return.
    if ($time) { $param['timestamp'] = $time; }
    else        { $param['timestamp'] = curl_call(API_URI . 'shoutlet_time'); }

    // Assign parameters to send.
    $method      = trim(strtolower($method));
    $param['mode'] = trim(strtolower($mode));
    $param['apikey'] = API_KEY;

    // Create a sha512 hash of the key, timestamp, secret and method endpoint. These
    // values MUST be in the proper order.
    $param['sha_hash'] = hash('sha512', API_KEY . $param['timestamp'] . API_SECRET .
    $method);

    // Initiate CURL, a POST call will always work (some methods do allow GET calls).
    $curl      = curl_init(API_URI . $method);
    curl_setopt($curl, CURLOPT_POST,      true);
    curl_setopt($curl, CURLOPT_POSTFIELDS, $param);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
    $return    = curl_exec($curl);
    curl_close($curl);

    // Return the data returned, format will vary depending upon mode.
    return $return;
}
```

## Listing Contest Fields

This method provides a means of listing special fields setup for a specific contest and is useful for working with contest data exports.

<b>Method</b>	contest_field_list
<b>SSL Required?</b>	yes
<b>POST Required?</b>	no
<b>CSV Output Mode?</b>	no

### *Additional Parameters*

**contest\_id (required string)**

The unique ID of the contest to list the fields of. Reference [get\\_contests](#) for active contest IDs.

### *Additional Results*

**field\_name (string)**

The unique name of the field for this account

**field\_alias (string)**

The "human readable" name for this field

**results (integer)**

The number of fields returned

## Exporting Contest Entries

This method allows you to retrieve all of the entries in a given contest. You can choose to have the data sorted on either the entry's date of creation or the most recent update date. You can specify the fields that you would like to have exported in the export.

<b>Method</b>	contest_entry_export
<b>SSL Required?</b>	yes
<b>POST Required?</b>	no
<b>CSV Output Mode?</b>	yes

### *Additional Parameters*

#### **contest\_id (required string)**

The unique ID of the contest that you wish to export. Reference [get\\_contests](#) for active contest IDs.

#### **page (integer optional defaults 1)**

By default the data is exported in paginated format, with 20 entries per page. If you wish to export a specific page beyond the first, you must specify it using this parameter.

#### **perpage (integer optional defaults 20)**

The number of entries to export in a single call, the minimum/default is 20, the maximum is 500.

#### **fields (string)**

A comma-separated list of the fields in each profile that are to be exported. If this is not specified, the results will default to exporting the e-mail, first name and last name. Reference the [contest\\_field\\_list](#) method for valid fields.

#### **include\_vote\_count (integer optional)**

This parameter is optional and should be included with a value of 1 if the return values should include the number of votes each entry has received.

#### **sort (string optional defaults "created")**

This parameter takes one of two values, "updated" or "created". These determine whether the data is sorted before output based on the date entries were created, or their most recent update date. If this parameter is not specified, the default sort order is the same as that specified by "created".

### *Additional Results*

**total (integer)**

The total number of entries returnable. Not available via CSV.

**vote\_count (integer)**

The number of votes that the entry has received. Will only be included if the "include\_vote\_count" parameter is included in the call.

**status (string)**

The status of the entry: "approved" or "denied".

**total\_pages (integer)**

The total number of pages returnable. Not available via CSV.

**page (integer)**

The current page returned. Not available via CSV.

**perpage (integer)**

The number of entries per page/call. Not available via CSV.

**results (integer)**

The number of entries returned. Not available via CSV.

**Entry data as specified by the field parameter.**

**Note:** Per the Twitter Terms of Service, automated exports of Twitter data can only include the Tweet ID and the User ID of the tweet's author.

## Listing Active Contests

This method lists all active contests for the account including reference information needed for the [add\\_contest\\_entry](#) method.

<b>Method</b>	get_contests
<b>SSL Required?</b>	yes
<b>POST Required?</b>	no
<b>CSV Output Mode?</b>	no

### *Additional Parameters*

**offset (optional integer defaults 0)**

The number of records to skip while paginating.

**per\_page (optional integer defaults 50)**

The number of records to return per call to a max of 100.

### *Additional Results*

**results (array)**

An array of contests which may include:

- **id (string)** - The unique ID of the contest
- **name (string)** - The name given to the contest
- **fields (array)** - The label and unique ID for all contest specific fields, the unique ID is used by [add\\_contest\\_entry](#)
- **files (array)** - The label, file type, folder ID and unique ID for all file upload fields for the contest, the unique ID is used by [add\\_contest\\_entry](#)
- **created (integer)** - The timestamp of the contest creation date and time

**total (integer)**

The total number of active contests.

**offset (integer)**

The number of contest records skipped while paginating.

**per\_page (integer)**

The maximum number of contest records returned while paginating.

**returned (integer)**

The number of contest records returned.

## Adding a Contest Entry

This method adds an entry to a contest allowing for external integrations into the contest engine.

<b>Method</b>	add_contest_entry
<b>SSL Required?</b>	yes
<b>POST Required?</b>	yes
<b>CSV Output Mode?</b>	no

### *Additional Parameters*

#### **contest\_id (required string)**

The unique ID of the contest to add the entry to. Reference the [get\\_contests](#) method for valid contest IDs.

#### **contest fields (optional string/file data)**

An arbitrary number of values referencing the unique ID from the field or file arrays of the [get\\_contests](#) results you can pass entry data into the contest.

#### **profile fields (optional string)**

An arbitrary number of values referencing the field\_name from the profile [field\\_list](#) results you can pass profile entry data into the contest.

### *Additional Results*

#### **entry\_id (string)**

The unique ID of the entry created.

#### **profile\_id (string)**

The unique ID of the profile created.

## Voting on a Contest Entry

This method adds a vote to a contest entry.

<b>Method</b>	contest_add_vote
<b>SSL Required?</b>	yes
<b>POST Required?</b>	yes
<b>CSV Output Mode?</b>	no
<b>Enterprise Account Required?</b>	no

### *Additional Parameters*

**contest\_id (required string)**

The unique ID of the contest holding the entry. Reference the [get\\_contests](#) method for valid contest IDs.

**entry\_id (required string)**

The entry ID to be voted upon.

### *Additional Results*

**success (boolean)**

Indicates success of the vote action.

**message (string)**

The vote success message as specified in your contest configuration.

## Voting on a Contest Entry with voting limits

This method adds a vote to a contest entry provided that the logged in Facebook user has not already voted in excess of set limits. The Facebook information is that which is provided by the Facebook Javascript SDK (<https://developers.facebook.com/docs/javascript>)

<b>Method</b>	contest_add_vote_fb_auth
<b>SSL Required?</b>	yes
<b>POST Required?</b>	yes
<b>CSV Output Mode?</b>	no
<b>Enterprise Account Required?</b>	no

### *Additional Parameters*

**contest\_id (required string)**

The unique ID of the contest holding the entry. Reference the [get\\_contests](#) method for valid contest IDs.

**entry\_id (required string)**

The entry ID to be voted upon.

**fb\_uid (required string)**

The Facebook User ID for the logged in user.

**fb\_token (required string)**

The access token provided to access information about the logged in user.

**fb\_signed (required string)**

The validation signature provided by the Facebook SDK.

### *Additional Results*

**success (boolean)**

Indicates success of the vote action.

**message (string)**

The vote success message as specified in your contest configuration.

## Updating a Contest Entry Status

This method changes the status of a single contest entry. It allows you to approve, reject, set to pending, or mark as winner.

<b>Method</b>	contest_entry_update_status
<b>SSL Required?</b>	yes
<b>POST Required?</b>	yes
<b>CSV Output Mode?</b>	no
<b>Enterprise Account Required?</b>	no

### *Additional Parameters*

**contest\_id (required string)**

The unique ID of the contest holding the entry. Reference the [get\\_contests](#) method for valid contest IDs.

**entry\_id (required string)**

The entry ID to be updated.

**status (required string)**

The new status of the entry. Can be one of ['pending', 'approved', 'denied', 'winner']

### *Additional Results*

**success (boolean)**

Indicates success of the status change action.

## Deleting a Contest Entry

This method deletes a contest entry.

<b>Method</b>	contest_entry_delete
<b>SSL Required?</b>	yes
<b>POST Required?</b>	yes
<b>CSV Output Mode?</b>	no
<b>Enterprise Account Required?</b>	no

### *Additional Parameters*

**contest\_id (required string)**

The unique ID of the contest holding the entry. Reference the [get\\_contests](#) method for valid contest IDs.

**entry\_id (required string)**

The entry ID to be deleted.

### *Additional Results*

**success (boolean)**

Indicates success of the delete action.

## Counting an Entry's Votes

This method adds an entry to a contest allowing for external integrations into the contest engine.

<b>Method</b>	contest_entry_vote_count
<b>SSL Required?</b>	yes
<b>POST Required?</b>	no
<b>CSV Output Mode?</b>	no
<b>Enterprise Account Required?</b>	no

### *Additional Parameters*

**contest\_id (required string)**

The unique ID of the contest to add the entry to. Reference the [get\\_contests](#) method for valid contest IDs.

**entry\_id (required string)**

The entry ID to retrieve votes for.

### *Additional Results*

**success (boolean)**

Indicates success of the vote action.

**count (integer)**

The number of votes for the entry.

## Obtaining the Server's Time

This method returns the current epoch time (aka UNIX timestamp) on the server at the time that the response was sent. This can be used to minimize problems with timeouts due to possible clock skew between your own server time and that on Shoutlet's servers.

<b>Method</b>	shoutlet_time
<b>SSL Required?</b>	yes
<b>POST Required?</b>	no
<b>CSV Output Mode?</b>	no

### *Parameters*

#### **None.**

This method can be called un-authenticated.

### *Results*

#### **Standard Results Do Not Apply**

The returned value is the current epoch time on the Shoutlet servers in plain text. This is the same value as provided by the "time" return value from most API calls.

## Testing Your API Connection

This method allows you to check if you can make valid connections to the Promotions API

<b>Method</b>	test
<b>SSL Required?</b>	yes
<b>POST Required?</b>	no
<b>CSV Output Mode?</b>	no

### *Additional Parameters*

**None.**

### *Additional Results*

**message (string)**

“test passed” on success

## Checking the Current API Version

This method allows you to find the current API version.

<b>Method</b>	get_api_version
<b>SSL Required?</b>	yes
<b>POST Required?</b>	no
<b>CSV Output Mode?</b>	no

### *Additional Parameters*

**None.**

### *Additional Results*

#### **version (string)**

The current API version number