



API Documentation

Version 2.0

9/2011

Covered Inside

- API Overview
- Methods & Examples
- Entity Definitions

API Overview

The FanTools API allows developers to perform functions within the FanTools administrative environment on behalf of any user. This includes creating and updating users, retrieving deal information, and retrieving an authenticated log on URL.

An API key is required for each organization within the FanTools application. This key is used to acquire access tokens, which are used as the authentication mechanism to perform actions on behalf of any user. All API methods require an access token authentication in order to perform the desired function. An API key can be obtained by contacting FanTools support.

The FanTools API is entirely HTTP-based. Methods to retrieve data generally require a GET request. Methods that submit or change data require a POST.

The API is a RESTful resource. The FanTools API attempts to conform to the design principles of Representational State Transfer (REST). The FanTools API supports only the JSON data format. For more information on REST, visit: <http://en.wikipedia.org/wiki/REST>.

Methods & Examples

GET token/access_token

Returns the FanTools API Response JSON object. The data property contains the access token for the requested user. This token should be stored for future reference. It will be used as authentication in all other API methods.

URL

http://www.fantools.net/token/access_token

Required Authentication

True

Parameters

api_key: The organization's api_key for the user the request will be fulfilled

Person: The Person entity. This is the user that will be granted the access_token

Example

```
$.ajax({
  type: 'GET',
  data: { Email: 'andrew@four51.com' },
  url:
'http://www.fantools.net/token/access_token?api_key=ee9ac02adfe87b06d6b1',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').html(response.data);
  }
});
```

GET list/deals

Returns the FanTools API Response JSON object. The data property contains an array of Deal entities. This method does not require authentication retrieving deals by Facebook page id. When requesting with access token, you must specify the org id of the company you are administering in the url. The deals returned will be all associated to that user.

URL

`http://www.fantools.net /api/list/{facebook page id}/deals`

`http://www.fantools.net/api/list/{org id}/deals?access_token={token}`

Requires Authentication

False

Parameters

None

Example

```
$.ajax({
  type: 'GET',
  url: 'http://www.fantools.net/api/list/1405066725585/Deals',
  dataType: 'jsonp',
  success: function(data) {
    $.each(data.data, function(i, deal) {
      $('#output').append(deal.Name + "<br />");
    });
  }
});
```

GET save/person

Returns the FanTools API Response JSON object. The data property contains the Person entity if the user creation or edit was successful. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/save/{org id}/person?access_token={token}`

Required Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'POST',
  data: { Email: 'jhein@four51.com', FirstName: 'John', LastName: 'Smith',
Password: '12345z', Phone: '612-555-8888', Zip: '55497', Roles: '100,245', IsSuperAdmin: false
},
  url:
'http://www.fantools.net/api/Save/1234/Person?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET save/role

Returns the FanTools API Response JSON object. The data property contains the role entity if the role creation or edit was successful. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/save/{org id}/role?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$ajax({
  type: 'POST',
  data: { Name: 'New Role', CreateEditDivisions: true, CreateEditClients: true,
CreateEditUsers: true, CreateEditRoles: true, CreateEditFacebookPages: true,
CreateEditTwitterAccounts: true, CreateEditFanGates: false, CreateEditStores: false,
CreateEditOffers: true, CreateEditTransactions: false},
  url:
'http://www.fantools.net/api/Save/1234/Role?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET list/roles

Returns the FanTools API Response JSON object. Returns an array of all roles associated with an org id. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/list/{org id}/roles?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'POST',
  data: {},
  url:
'http://www.fantools.net/api/list/1234/roles?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET list/facebookpages

Returns the FanTools API Response JSON object. Returns an array of all Facebook Pages associated with an org id. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/list/{org id}/facebookpages?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'POST',
  data: { },
  url:
'http://www.fantools.net/api/list/1234/facebookpages?access_token=810ae8cbc8b246aabe13a8
',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET list/twitteraccounts

Returns the FanTools API Response JSON object. Returns an array of all Facebook Pages associated with an org id. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/list/{org id}/twitteraccounts?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'POST',
  data: { },
  url:
'http://www.fantools.net/api/list/1234/twitteraccounts?access_token=810ae8cbc8b246aabe13a8'
,
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET list/locations

Returns the FanTools API Response JSON object. Returns an array of all Facebook Pages associated with an org id. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/list/{org id}/locations?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'POST',
  data: { },
  url:
'http://www.fantools.net/api/list/1234/locations?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET signon

Returns the FanTools API Response JSON object. The data property contains a URL for the automatic logon. The URL contains a querystring with a temporary token that expires in 5 minutes after issuance. The requesting user must belong to the same organization as the target user.

URL

`http://www.fantools.net/api/signon?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'GET',
  data: { Email: 'andrew@four51.com' },
  url:
'http://www.fantools.net/api/SignOn?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('a').attr('href', response.data);
  }
});
```

GET save/client - Update

Returns the FanTools API Response JSON object. The data property contains the client entity if the client creation or edit was successful. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/save/{org id}/client?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$ajax({
  type: 'GET',
  data: { Name: 'New Organization'},
  url:
'http://www.fantools.net/api/1234/client?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET save/division

Returns the FanTools API Response JSON object. The data property contains the division entity if the division creation or edit was successful. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/save/{org id}/division?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'GET',
  data: { Name: 'New Organization', ParentID: '100' },
  url:
'http://www.fantools.net/api/1234/division?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

GET save/offer

Returns the FanTools API Response JSON object. The data property contains the deal entity if created properly. You must specify the org id of the company you are administering in the url.

URL

`http://www.fantools.net/api/save/{org id}/offer?access_token={token}`

Require Authentication

True

Parameters

access_token

Example

```
$.ajax({
  type: 'GET',
  data: { Name: 'September Coupon',
    Type: 'Coupon',
    FacebookPages: '87614383,138768766',
    Headline: 'Buy One Get One Free!',
    PublishDate: '10/26/2011 10:00 AM'
  },
  url:
  'http://www.fantools.net/api/1234/division?access_token=810ae8cbc8b246aabe13a8',
  dataType: 'jsonp',
  success: function(response) {
    $('#output').append(JSON.stringify(response.data));
  }
});
```

Entity Definitions

Response

The formatted response for every api method call. Even on errors the response returned is Http Status Code 200. The error message will detail the cause of the error and the success property will be false.

success: (boolean) True when calls was successful. False when an error was encountered

message: (string) Informative message in response to the method call

data: (object) The entity in relation to the method call

JSON example:

```
{
  success: true,
  message: 'The url provided is the auto sign on entrance point. It will expire in 5
minutes.',
  data:
'http://www.fantools.net/Admin/Login/Auto?email=user@four51.com&token=
84d329db1dfe87b06d6b1'
}
```

Person

The object representing an administrative user in FanTools.

Email: (string) Email address and username

FirstName: (string)

LastName: (string)

Phone: (string)

Zip: (string)

IsSuperAdmin: (boolean) Indicates whether user is a Super Admin.

Roles: (string) Comma separated list of roles assigned to user.

Role

The object representing an a role that can be assigned to admin users in FanTools.

ID: (string) Unique identifier for the role

Name: (string) The name of the role

CreateEditDivisions: (bool) Indicates whether the role has the Create/Edit Division permission

CreateEditClients: (bool) Indicates whether the role has the Create/Edit Client permission

CreateEditUsers: (bool) Indicates whether the role has the Create/Edit User permission

CreateEditRoles: (bool) Indicates whether the role has the Create/Edit Roles permission

CreateEditFacebookPages: (bool) Indicates whether the role has the Create/Edit Facebook Pages permission

CreateEditTwitterAccounts: (bool) Indicates whether the role has the Create/Edit Twitter Accounts permission

CreateEditFanGates: (bool) Indicates whether the role has the Create/Edit FanGates permission

CreateEditStores: (bool) Indicates whether the role has the Create/Edit Locations permission

CreateEditSurveys: (bool) Indicates whether the role has the Create/Edit Surveys permission

CreateEditOffers: (bool) Indicates whether the role has the Create/Edit Promotions permission

CreateEditTransactions: (bool) Indicates whether the role has the Create/Edit Transactions permission

Client

The object representing a client.

ID: (string) Unique identifier for the client

Name: (string) The name of the client

ApiKey: (string) ApiKey for the client

ParentID: (string) The unique identifier for the parent organization.

Division

The object representing a division of an organization.

ID: (string) Unique identifier for the division

Name: (string) The name of the division

ApiKey: (string) ApiKey for the division

ParentID: (string) The unique identifier for the parent division.

FacebookPages

The object representing the details for Facebook Pages.

ID: (string) Comma-separated list of Facebook Pages using Facebook's ID for each Page.

TwitterAccounts

The object representing the details for Twitter accounts.

ID: (string) Comma-separated list of Twitter accounts using Twitter's ID for each account.

Locations

The object representing locations.

ID: (string) Comma-separated list of Locations using FanTools ID for each location.

Deals

The object representing the details for deals. These are the visible properties of a deal. It does not include supportive objects such as profile questions or fan gates.

ID: (string) A unique id for the deal

Type: (string) Coupon, Group, Flash or Post

Name: (string) The title of the deal.

FacebookPages (string) A comma-separated list of the Facebook Page ID's to which the deal is published

TwitterAccounts (string) A comma-separated list of the Twitter Account ID's to which the deal is published

Headline: (string) The description that appears in Facebook and Twitter

ShortDescription: (string)

Description: (string) HTML formatted string.

Restrictions: (string)

CouponCode: (string)

Url: (string) Buy Now Url

MinUnits: (string) For group deals, the minimum number of deals that must be purchased before the deal tips

MaxUnits: (string) For group deals, the maximum number of deals that can be purchased.

MaxIndividualUnits: (string) For group deals, the maximum units one person can purchase.

Price: (string) For group deals, the price of each unit.

Value: (string) For group deals, the value of each unit.

PublishDate: (string) Ex: 2011-05-13 12:00:00

StartDate: (string) Ex: 2011-05-13 12:00:00

EndDate: (string) Ex: 2011-05-13 12:00:00

ExpirationDate: (string) Ex: 2011-05-13 12:00:00

PublishUrl: (string) Url that links to deal on Facebook

DesktopLayout: (string) HTML formatted string

MobileLayout: (string) HTML formatted string

PublishToWall: (boolean)

WallPostMessage: (string)

Locations: (string) A comma-separated list of locations assigned to the deal

Tweet: (string)

EmailSubject: (string)

EmailBody: (string)

SMSMessage: (string)

ImageURL: (string) Url to the deal image

DateCreated: (string) Ex: 2011-05-13 12:00:00

NotificationsSent: (boolean) Indicates all broadcasts (SMS, Email, Wall Post) were processed.