socialize.getUserInfo JS

Description

This method is used to retrieve extended information regarding the current user. The response of this method includes a User object enfolding user data. See the User object page for the full specification of the available information.

Note: This method is also supported in our REST API. If you wish to execute this method from your server, please refer to REST API > socialize.getUserInfo.

Securing the getUserInfo Process

The Gigya service supports a mechanism to verify the authenticity of the getUserInfo process.
If the global configuration object's **signIDs** field is set to 'true', Gigya "signs" the *UID* field received in the response with a cryptographic signature to prevent fraud.
The User object that you receive in the response has a cryptographic signature (*UIDSignature* data member) provided by Gigya.
We highly recommend verifying the authenticity of the signature to prove that it has indeed originated from Gigya rather than somewhere else.
Please follow the instruction in the Signature Validation Process section in the Security page of the Developer's Guide.

> **Note:** The signature mechanism prevents any tampering with the user ID (UID) but it is not meant to protect other user data fields. For retrieving sensitive data (such as email addresses) we recommend using the REST API > socialize.getUserInfo call to Gigya directly from your server to retrieve that data.

Syntax

Parameters

The following table lists the available parameters:

Required	Name	Type	Description
	callback	function	A reference to a callback function. Gigya calls the specified function along with the results of the API method when the API method completes. The callback function should be defined with the following signature: ***functionName(*****Response*****)***. The "Response Object Data Members" table below provides specification of the data that is passed to the callback function.
	cid	string	A string of maximum 100 characters length. The CID sets categories for transactions that can be used later for filtering reports generated by Gigya in the "Context ID" combo box. The CID allows you to associate the report information with your own internal data. For example, to identify a specific widget or page on your site/application. You should not define more than 100 different context IDs. **Note:** This parameter overrides the value of the identical parameter in Global Conf (the global configuration object). If the parameter is not set for the method, the value from Global Conf is used.
	includeAllIdentities	Boolean	This parameter's default value is 'false'. If set to 'true' you will receive all the user's identities, including those with expired sessions. The identities are received in the response: *response.user.identities* is an array of Identity objects. Each object has an attribute *response.user.identities[i].isExpiredSession* that is 'true'when the session expired for that provider (or is otherwise inactive) and 'false'if it is active.
	context	object	A developer-created object that is passed back unchanged to the application as one of the fields in the response object.

	extraFields	string	This parameter accepts a comma-separated list of additional data fields to retrieve. The current valid values are: ***languages, address, phones, education, honors, publications, patents, certifications, professionalHeadline, bio, industry, specialties, work, skills, religion, politicalView, interestedIn, relationshipStatus, hometown, favorites, likes, followersCount, followingCount, name, username, educationLevel, locale, verified, irank, timezone,*** and ***samlData***. **Note:** Before your application can retrieve Facebook data, the user must grant your application with access. Please make sure you have checked the check boxes that enable retrieving the relevant fields from Facebook in the Permissions page on Gigya's website. You may find more information in the Facebook Permissions section of our guide.
	enabledProviders	string	A comma delimited list of providers from which to return user information in the response.
	disabledProviders	string	A comma delimited list of providers from which not to return user information in the response.
	signIDs	Boolean	The default value is 'false'. When this field is set to 'true', the User object returned by the method is signed by Gigya. To learn more about this subject, please refer to the Security page of the Developer's Guide. **Note:** This parameter overrides the value of the identical parameter in Global Conf (the global configuration object). If the parameter is not set for the method, the value from Global Conf is used.
	includeiRank	Boolean	***Deprecated*** . This parameter's default value is 'false'. If set to 'true' you will receive the user's iRank (influencer rank) in the response User object.

Response Object Data Members

Field	Type	Description
errorCode	integer	The result code of the operation. Code '0' indicates success, any other number indicates failure. For a complete list of error codes, see the Error Codes table.
errorMessage	string	A short textual description of an error, associated with the errorCode, for logging purposes. This field will appear in the response only in case of an error.
errorDetails	string	This field will appear in the response only in case of an error and will contain the exception info, if available.
fullEventName	string	The full name of the event that triggered the response. This is an internally used parameter that is not always returned and **should not** be relied upon by your implementation.
callId	string	Unique identifier of the transaction, for debugging purposes.
time	string	The time of the response represented in ISO 8601 format, i.e., yyyy-mm-dd-Thh:MM:ss.SSSZ or
statusCode	integer	The HTTP response code of the operation. Code '200' indicates success. This property is deprecated and only returned for backward compatibility.
statusReason	string	A brief explanation of the status code. This property is deprecated and only returned for backward compatibility.
user	User object	A User object with updated information for the current user.

Code Sample

```
function printResponse(response) {
    if ( response.errorCode == 0 ) {
        var user = response['user'];
        var msg = 'User '+user['nickname'] + ' is ' +user['age'] + ' years
old';
        alert(msg);
    }
    else {
        alert('Error :' + response.errorMessage);
    }
}

gigya.socialize.getUserInfo({callback:printResponse});
```

Notes:

- This sample is not meant to be fully functional code. For brevity's sake, only the code required for demonstrating the API call itself is presented.
- To run the code on your own domain, add your Gigya API key to the gigya.js URL. A Gigya API key can be obtained on the Site Dashboard page on Gigya's website. Please make sure that the domain from which you are loading the page is the same domain name that you used for generating the API key.
- In some cases it is necessary to **connect/login** the user to a provider ? prior to calling the API method. You can learn more in the Social Login guide.

In the Get User Information Demo page you will find a complete working example that uses socialize.getUserInfo method. You may view the code, run it and view the outcome.