**MembersMapper Service Api Documenation (Members area)**

**Descritipon :**

Returns collections of things like likes as well as search data results and mapps them to view models that can be used to display the user search results. Eventually this code will be moved to a search service.

Help URL:

<http://localhost/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/help>

**AuthRequired: Username password and Api Key**

\*\* request a correct API key from the service admins \*\*

A user authorization header and API key are needed in the header

Example of the authorization header for a user with **username** : case **password** : drive333

1.)Combine the username and password with a “:” separator - **> case:drive333**

2.) Url encode the combined string as base 64 : Y2FzZTpkcml2ZTMzMw==

i.e url for testing encoding of passwords

<http://www.motobit.com/util/base64-decoder-encoder.asp>

3.) Add the encoded URL to the header

Two headers are required one named “apikey” and one named “Authorization”

a)Sample of api key and authentication header using fiddler :

User-Agent: Fiddler

Content-Type: text/json

Host: localhost

Authorization: Y2FzZTpkcml2ZTMzMw==

apikey: 460ad6f3-8216-469f-9b1c-52cffa5d812c

b) Sample of Ajax calls with header included:

<http://wilbloodworth.com/add-request-headers-in-ajax-http-post-using-jquery/>

**Service Call Details and Documenation**

Most of the models that pass values use this viewmodel to hold the properties :

{

"activationcode":"String content",

"email":"String content",

"openididentifier":"String content",

"openidprovider":"String content",

"password":"String content",

"photoid":"1627aea5-8e0a-4371-9022-9b504344e724",

"profileid":2147483647,

"screenname":"String content",

"securityanswer":"String content",

"securityquestion":"String content",

"sessionid":"String content",

"username":"String content"

}

The following details of each call shows the properties required for each call on a call by call basis.

Matches and Such

1)**getwhoilikecount** – gets the number of profile likes by profile id

**Help url** :

<http://localhost/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/help/operations/getquickmatches>

**Url** :

<http://localhost/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/getquickmatches>

Json Body

{

**"profileid":12**

}

Profile id is required

Result:

A JSON encoded list of List<MemberSearchViewModel>

Ordered by creation date and has photos : means that the top profiles that have photos will will show up. 50 profiles are returned max.

Here is the format.

[{

"MyCatchyIntroLineQuickSearch":"String content",

"ProfileVisibility":true,

"aboutme":"String content",

"age":2147483647,

"birthdate":"\/Date(928167600000-0500)\/",

"blockdate":"\/Date(928167600000-0500)\/",

"city":"String content",

"countryid":2147483647,

"countryname":"String content",

"creationdate":"\/Date(928167600000-0500)\/",

"distancefromme":1.26743233E+15,

"galleryphoto":{

"convertedsize":9223372036854775807,

"creationdate":"\/Date(928167600000-0500)\/",

"description":"String content",

"imagecaption":"String content",

"imagename":"String content",

"orginalsize":9223372036854775807,

"photo":[81,

109,

70,

122,

90,

83,

65,

50,

78,

67,

66,

84,

100,

72,

74,

108,

89,

87,

48,

61],

"photoid":"1627aea5-8e0a-4371-9022-9b504344e724",

"profileid":2147483647,

"screenname":"String content"

},

"genderid":2147483647,

"hasgalleryphoto":true,

"id":2147483647,

"interestdate":"\/Date(928167600000-0500)\/",

"lastloggedonstring":"String content",

"lastlogindate":"\/Date(928167600000-0500)\/",

"latitude":1.26743233E+15,

"likedate":"\/Date(928167600000-0500)\/",

"longitude":1.26743233E+15,

"lookingForageto":"String content",

"lookingforagefrom":"String content",

"lookingforid":2147483647,

"mycatchyintroline":"String content",

"online":true,

"peekdate":"\/Date(928167600000-0500)\/",

"postalcode":"String content",

"postalcodestatus":true,

"profilephotos":{

"ProfilePhotosApproved":[{

"convertedsize":9223372036854775807,

"creationdate":"\/Date(928167600000-0500)\/",

"description":"String content",

"imagecaption":"String content",

"imagename":"String content",

"orginalsize":9223372036854775807,

"photo":[81,

109,

70,

122,

90,

83,

65,

50,

78,

67,

66,

84,

100,

72,

74,

108,

89,

87,

48,

61],

"photoid":"1627aea5-8e0a-4371-9022-9b504344e724",

"profileid":2147483647,

"screenname":"String content"

}],

"ProfilePhotosNotApproved":[{

"convertedsize":9223372036854775807,

"creationdate":"\/Date(928167600000-0500)\/",

"description":"String content",

"imagecaption":"String content",

"imagename":"String content",

"orginalsize":9223372036854775807,

"photo":[81,

109,

70,

122,

90,

83,

65,

50,

78,

67,

66,

84,

100,

72,

74,

108,

89,

87,

48,

61],

"photoid":"1627aea5-8e0a-4371-9022-9b504344e724",

"profileid":2147483647,

"screenname":"String content"

}],

"ProfilePhotosPrivate":[{

"convertedsize":9223372036854775807,

"creationdate":"\/Date(928167600000-0500)\/",

"description":"String content",

"imagecaption":"String content",

"imagename":"String content",

"orginalsize":9223372036854775807,

"photo":[81,

109,

70,

122,

90,

83,

65,

50,

78,

67,

66,

84,

100,

72,

74,

108,

89,

87,

48,

61],

"photoid":"1627aea5-8e0a-4371-9022-9b504344e724",

"profileid":2147483647,

"screenname":"String content"

}],

"SingleProfilePhoto":{

"convertedsize":9223372036854775807,

"creationdate":"\/Date(928167600000-0500)\/",

"description":"String content",

"imagecaption":"String content",

"imagename":"String content",

"orginalsize":9223372036854775807,

"photo":[81,

109,

70,

122,

90,

83,

65,

50,

78,

67,

66,

84,

100,

72,

74,

108,

89,

87,

48,

61],

"photoid":"1627aea5-8e0a-4371-9022-9b504344e724",

"profileid":2147483647,

"screenname":"String content"

}

},

"screenname":"String content",

"stateprovince":"String content"

}]