**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://173.160.122.195/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/help](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: 173.160.122.195

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**

There are two types of matches provided by the service : Email matches which are meant for email dilievery and Normal matches which are used by mobile and web clients to display matches.

Mobile/Web Matches

Matches For web and mobile applications.

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

**Help url** :

[http://173.160.122.195/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/help/operations/getquickmatches](http://localhost/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/help/operations/getquickmatches)

**Url** :

[http://173.160.122.195/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/getquickmatches](http://localhost/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/getquickmatches)

Only the profile ID is needed as the auth infor by either API key or UserName password has been already sent.

Json Body

{

**"profileid":1**

}

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":”base64 string”,

,

"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":”base64 string”,

"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":”base64 string”,

"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":[],

"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":[],

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

"profileid":2147483647,

"screenname":"String content"

}

},

"screenname":"String content",

"stateprovince":"String content"

}]

Important Elements to note :

4 batches of phots are passed down **\*\* Code change all the photos are base64 strings \*\*** and wil need to be converted on the client.

Galleryphoto – which is just a single photo and is the photo that should be displayed to identify the user.

* If gallery photo is empty a black image of a male or female should be displayed

Other photo sets are lists and are are contained in the ProfilePhotos object.

ProfilePhotosApproved : a list of all the PUBLIC approved photos – this info is needed for the profile detail. It may or may not be passed down from the get quick matches and may need to be re-retrived on the next page i.e if a profile is clicked on.

ProfilePhotosNotApproved: will not be populated by this call used for admin page purposes only.

ProfilePhotosPrivate : A list PRIVATE approved photos. The user settings are used to determine who and who cannot view this list of photos.

SingleProfilePhoto : A single photo which is also a duplicate of the gallery photo , you can use that if the gallery photo is empty.

Last Logged In String : a formatted bit of text that shows when the user was last logged in.

hasGalleryPhoto : a quick way to check if a profile has a main or gallery photo.

Online : shows on most screens , tells the UI that the user is currently online , if so chat lights should also blink if the chat API is available.

All other fields are self evident and further documentation can be provided as needed.

Sample call

[http://173.160.122.195/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/getquickmatches](http://localhost/Shell.MVC2.Web.MembersService/MembersMapperService.svc/Rest/getquickmatches)

Json Body

{

**"profileid":1**

}

Json Result: An array of the following objects

In this case 2 profiles are retured as results

[{"MyCatchyIntroLineQuickSearch":"Hi There","ProfileVisibility":null,"aboutme":null,"age":42,"birthdate":"\/Date(19544400000-0500)\/","blockdate":null,"city":null,"countryid":72,"countryname":null,"creationdate":"\/Date(1329312476527-0600)\/","distancefromme":149.8591577994286,"galleryphoto":{"convertedsize":3487,"creationdate":"\/Date(1328124657800-0600)\/","description":null,"imagecaption":"070.jpg","imagename":null,"orginalsize":677756,"photo":"\/9j\/4AAQSkZJRgABAQEAYABgAAD\/2wBDAAQCAwMDAgQDAwMEBAQEBQkGBQUFBQsICAYJDQsNDQ0LDAwOEBQRDg8TDwwMEhgSExUWFxcXDhEZGxkWGhQWFxb\/2wBDAQQEBAUFBQoGBgoWDwwPFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhb\/wAARCABfAEcDASIAAhEBAxEB\/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL\/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6\/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL\/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6\/9oADAMBAAIRAxEAPwDy1dVjWJYYVWONeijvVa41Jj0zXELrhdVkimWRX+6y8g1Ja6m9z8wk3AMVyDnkckfWuz2sOxxJPqdBq1zcz2M0cRyzoQuT3rnbOwvLeEK8kW7uA3SrBnf+J8fVsUxrjnG+P8xUSkty9yWOC5bG6ZF59asBrqPpcxn2zWdM8u3KMhPYZFbMOmpGsbwXNnqLMmZA9wsYQkA4AJB49acZpdQsuxC19fLwBG3HZhUU2oXpX5oX59MGnahG8K+eNMiWJMh\/LlEnP4NkUmmz2E8OZLe4DY+8CVX8Ac1rGd9pDUV2K\/2mduXik\/75NFXIUULliV+hoo9mn1KUDznwbp9xPq94LSZlhV9gU8q7E9SPoK6630jVby8XTdPs7m8upflSK1iLsT7Acmq3gfRNSPg9L2wLRvJKXLgcjPA\/QfrU\/h9\/HOk+JY7rQNf1TTtQnP2dbi0naOTDkAjK9jx+VZfV58t7bmSqRk7X1KfiXwrr3h68ktdctLizu4wGa2nQq6AjILKeRwRx71j3FhqEl1DZQrK9zMcpGj4b1wcd\/YV9A+IfBep634qt7GSee+vpZESe5uZDI8z4ALMx5J4yTX1T8K\/2e\/A2nafaT3OlWt5qEaBmupk+ZWxzt\/uiuV3UrM7\/AGa5L9T829HsLu6szLHNJ+7OHVzkqfrXZx\/C\/wCJC2kN1H4Y1WaC4jEkMsVu0iyIejKVByK+pv2ifgL4U8G67Z614fslsbfUXa3uYIiTGXIypwenINbP7NNybON\/Cd3cKIog09i7tgxjPzxg+mcN+dTKbjG63M61KSXMkfF+p6F4n0gj+0dJvbU9AbiBowfxYCobGDW7u6+zxyNG\/ZVySfp619S\/t3fEDwvYfDO\/8DRa7Dca1fyQq1qIpHEUBbezhyNoIwvGc89K+Vfgnb+KvFPxO0zwtod5sk1a9W3iMhT93GT8zbm6YQE\/hW1KTcbyRlG7V7G\/rnh7xbpHg\/8A4SO8sJ20tbkWxvCBt80gkL\/46fpRX0v+2j4Z1HQv2Z5Umso7eyt72zhhVZUKrtLAYAOent70VGHrTqw5pK2ppHVannf7PekabJ8NFgvWjX5oxk+0a\/1JrQ1rTfDmlagl5aTRy3MDhk6YDZ4J\/HFeRN4g1bRNJaxt7K6igD7o1C8kEDqPwqPVb+aPR9OmF1NJNNc770j7iEYKRj1wRz9RXo4msqVPlvrscOCwsq1bn6bn0X4Dvd2rR3Gn20t5e2f+lSxwR75Gj5XcAevIBr6G8A\/E3VNOtYV8R+F51tbmVIbWSwgMtw7tnCtEpO7ofu9O4rxT9g3SLPVNSv8AV9SgS6ikKWq5JDRFVGMEcjPP519PXXhvStP8UaXe2OlTNHEW3TK7zNGSD\/eYkA9yK82FRS1aPenTjF8up4\/8fvG154s8H3FrbeF547G7hWXTrgxFpt\/VHZcjaAcEgZIGc968Rg8Vf2Vp9n4n0Nbea7XMyQSvmMvt2vExHPBJH0wa+tvHGkeHtD8B61rdzaXcSWdvdXIjW6kjRwFZ8bA2Bk9uOtfm\/wCA7bxJdWbXNm811BeSy3V3Hv3GGbAbcM+oOMd8e1FTls7I3p04yXL0OQ+K3ibWfGPji\/1zVdMhtbi4mYtbqzFYjwMAnnHFZngfWb3wr440nxFDaxtJpd5HcqisV3bTnGe2eleh+N4NMurXTtWIAkmiKXUcQB2urFePY4B5rhNdn0oTvHBcowBIDYP512UFSqQ0Z5WIozovla0PQ\/jp+0Rr3xI8K\/8ACKzeHrSxsTcpcGT7S8khZQcDpgDk\/kKK8bdvOuBHbq0rekakk++KK0p0YRVoqyMVZaH0\/r0Xhy0hN9FZ295eSTGGGC4cBFAA\/eEHr6Aep74rklv7ea2vbG70mG2smhOzcQ5DAknbgcNnn36Ut46TxtG6Ix\/i3cEn2I6GsvVJI1kaRZSvmoFdCud2QeeOh4\/8dPtW2JwqmryepjgsTKlZJHuP7Beo6la+PdU8P2N159rd2hvYYHRkmYIQMqrAfMATwfvBfavr3T7O916Rru31R0ZF2NF5zoFP+6Oh\/Cvzy\/Z28Sy+BviRonibEzf2fdBp1B+eSEkh157lGbrX3+nxJ8ParpsesaT4R8Qal50e+N003ylbnA3SuyqBnjOT+NeXPDyo8vmetHFOcm0jw\/8Abr8ev4N8Lr8OdKvje614idDeKJiwtrYn7pJPytJjaOnG4+lfHPgi\/wBZ0caxo7b45pY3LQiQoXI6oe4I647gV7N+1doWvj4yXniLXY4vO1tQ8i225obMqSkcIYgZ2qmM4GSGOMYryi6s7V5EkcN52SWbdyoBG38etdUcGqtK6e4\/rsqdTVHY2\/gLw7rPg+HxB\/a\/9kq8UYuldQIFkC4bBPTJ5xXinxJk0208RXVro1ybiyjbZFPjHmYxkj2rsvF2rLYeGY9OjtQpuZMyyBFYT4OdhzyjY7j36V5Jrl3HNdyPzGrNkIDhU9h7VNGnUo3jJmeMrUqlpR3NXwOGvNUmhUt0zx1PtRUXwuuli8SZUg5Vh+hor08MoShdnlybuepX8sVvcNNG0ibjlwGyB77T\/SkEongEhcI4YbSRnnJ4\/P8AnXoX7Rvw70rwl4Xg8QaPcXccTSJFNaXeWY7hwwb+8D1B7fSvHNJuZLma0tFcgNdYY55AXLY\/Sso1oy2J9m4uzR1mmyPJIXZApRsMVPyv7j0+nrX1h8E\/Fl1qnwJgFlqckOr6DI8YbzCq7AQVDHocqwxnqV+tfKcChVxjAHQV6V+zf4gGmeNP7MuZ1js9VTypN+NvmDJQnPbOQfrRiqbnS03WpvhanJU16nrXxLuJviH8NdY07UoRFqumok8bIgyzKM9fRjt\/M181eUssau8as2MHI5r7R+F\/hR5Ncle6sEmWQSKLpHBMYI4jYHnHpj+Rr5N8YaRLonjDVtHnA3Wt3InCFeAxxweRxis8vl8UfmdGMvpI4jxPYyyaVdLZ+WtyIjJayTIHETgHkZB54xn3p2j+G9L8QeFrHW49A8OBryBJHU6naREPjDDYzgrhgRhgK9G8JfDPxJ4o8Ny+ILaK3i0e3Zrea6lkztk\/u7ByM5Ubjgcjmvl3xlo1\/oHiDUNLuQ0UtjdSQyxjOMqxGR7cV01KsYT93XucsoycU3sdB478Njw7qiapbxWtqGJSWO3v7aZSSDghInJXjrxjvxRXBTXUaMpZ2LfTp9aK5JYiKehUVK2x9t\/tsTzReFrWwHlLDJckGMKGIAyVwTyPumvmvwyrW+tWoZdsUVyJ5Hc44\/uj1zXqPxp8Uaj4u8QGe+MXl2jMiJGm0c89+pHT864+1t1fK7F3A8jHWqpYeSSbM6lZOTSN10C\/dO5T90\/j\/PipbGaS2ljmjYq0bBlYdQQc5qLS0SOGKLd+5mUbT\/cfGM49DjB+gqeeExMyN8rKcEV3R1WpLidN4v8AjT8Qb7Vbny9abTRNEInWwXy\/NTAGSTk5I64rm\/DN7d6jBd3N7czXFx9rffNPIXkcsFPLHk9ayvE223t4bx+Ah2Mcc4PI\/kfzrY8G2htNHjkmI86+mMwGcgZAxk+ygE1nTpqE7RRdScpQ1Pff2cfGuieFPh7r1hrs9qYby5Ux2kzqWuCI+QqE8nODx1xivlv9rDR9LvPitL4itrO+0zTtcXeBNCV3SKAGIJAzkbTwAASRXr37Hcf\/AAkXjDWdVu7G1mkF59kt5lcrNBGmfu5XHOc9Qai\/b6mF34W0uG4XzJbXUn+zXBGHaLYVYOOm7IXpxXDKzrSdtzstegk+h8garo8YmZLWYvtPDHo1FbCou77xHHaiiWHi3fY4Y4iSVj\/\/2Q==","photoid":"f6e7ddf2-96d3-4b38-bc7a-536ca6498454","profileid":3265,"screenname":"darnzky"},"genderid":2,"hasgalleryphoto":true,"id":3265,"interestdate":null,"lastloggedonstring":"Over one Month","lastlogindate":"\/Date(1329312476527-0600)\/","latitude":33.871944427490234,"likedate":null,"longitude":35.509723663330078,"lookingForageto":null,"lookingforagefrom":null,"lookingforid":null,"mycatchyintroline":null,"online":false,"peekdate":null,"postalcode":"","postalcodestatus":false,"profilephotos":null,"screenname":"darnzky","stateprovince":""},{"MyCatchyIntroLineQuickSearch":"Hi There","ProfileVisibility":null,"aboutme":null,"age":45,"birthdate":"\/Date(-48798000000-0500)\/","blockdate":null,"city":null,"countryid":67,"countryname":null,"creationdate":"\/Date(1327192022020-0600)\/","distancefromme":58.952548979832052,"galleryphoto":{"convertedsize":1912,"creationdate":"\/Date(1327192076437-0600)\/","description":null,"imagecaption":"34489\_1321387994876\_1236111684\_741878\_4574836\_n (1).jpg","imagename":null,"orginalsize":15591,"photo":"\/9j\/4AAQSkZJRgABAQEAYABgAAD\/2wBDAAQCAwMDAgQDAwMEBAQEBQkGBQUFBQsICAYJDQsNDQ0LDAwOEBQRDg8TDwwMEhgSExUWFxcXDhEZGxkWGhQWFxb\/2wBDAQQEBAUFBQoGBgoWDwwPFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhYWFhb\/wAARCABfADgDASIAAhEBAxEB\/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL\/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6\/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL\/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6\/9oADAMBAAIRAxEAPwD53Ye9QsMVYcCopCBW3MTykDLUckRVc9qmkI21FI5K4Y8UcwuUglA21BItTyEHpUDNjijmFykDR\/5NFJJJ2oouFkdC0ynPI4GeTVeSZfWvP\/8AhINSj5a9duejAH+lNm8Rag67TcsB\/sqB+tZXNbncX12lvayXEh+SNSTiudhk8TaqWnshsQD7sW0qOM5BPX3rO8NmLW7y5tb+7kjZbZ5LfMp+dxzj8q9Y8F6VNptnb3rJG9rIvzIep4JPHY8evNcmJxXslpud2Dwft3d7eRwOn39yWkttRjEF1CAW7KVPcVBea7p8anE+4rxhRkmq\/wAXCy61HMUVUlBKBc4x269a5HzFAGT37V0U6jlBM5K9NU6korobd74iuZGxAixj1PzGisFpCW9u1FXdmVkO8wlck5p6tk4\/KmRoVCuyM0ZbB5xn2zWp4X0LUvEOtR6XolhPe3cp+SGJdxx6k9AB6nFIYzw+C2vWcZ2\/PMq4PUZIr3mznnttGlgkn8k25PyjHJI4YHrjHpXefB34DS6f8N7rTNYvbVNZ1OZJDLHCHW3C8rEWIyQTyWGOcduvQ6L8FNautQjttYmtYLOE5kuIZA7yDphB2PuenvXDi6NSco2jc9bLK9KlCbnKzPjn4mS6jc+Jma\/t5oGSJBCs0ZQtHj5XA9G6g1zTAj86\/QT4ofBXwN4r0+1sLrSXhfT4Bb2lzbSFZYox0Ut\/FySfmzyTXgXxN\/ZW8R6Npt1qXhvVYdXgt0aQ2ssfl3BAGSFxlWOO3Fd6p8sUkeVOo5ycm9WfOxBoqzJEsce592fTbj+dFIQq7tvlY+XORX3j+xh4P0rRvgvo9\/FZRx3+rWv2y6m2gvLuYlQT1wF24HavhJSN3Sv0L\/Zlukk+BPgu5H+r\/suOF\/YjcvP5VUdxM9EjjjDlI4l\/AVdNuo2tj5hgZAqjo8rNcSW7r80J5PqOxrVz8w+taoTKU1spXOMY7VFLLGtuqbQT5pGMdcY\/xrQuCPLK98cVzeh3X2i9kf7xRcqM8BieuPbFHUD88f2iNLi0H4zeJtIt4gsFrqcoiA6BWbcAPoGorq\/23tOFn+0TrLLHtS7ht7hT2bMQBP5qaKyluylseUsqhuM47cV96\/sssIP2c\/DC3OPLexPU9QXY18G7GzXbfB6DUdd8aaToAv7p4J7hYI7Vrx1QbzglVHHAJOOPrU83KrlRhzNLY\/QXwm8jrcXExJ+cKrH+JFUYP45JrdjfL5J+6K5\/S3ht4UtLcBYYyQqjoqJhVH6AfhWtbyFUHq1aozHapOI\/LbsWx+deB\/F34p6v8OfF8lnYWFpdRzKXInLD+I4AINe1eKJ9lizj+Ahh+dfHf7c2o3UfxXjtolCxvpsciyH\/AGiwOPyqKnNbRmlJxUveV0eafHX4gX\/xK8cHxFqVja2Ui26WqRWzMV2oWwSW5ycmiuQlGOKKm9xaFjbxmvW\/2ONOST4sSa5KNyaDYS3KL\/elfEcY\/Nj+VeTtnpt5r2j9jGbZ4s1WFlG2W3hcn\/cckD88flSW4M+t\/D9rNBaIZ3VpG5bJzjJJx+ZNawn29SM9q5+xvgsPJI454p0mqqEOAzY7YroS0My14ouAbB13cspxz1NfJ\/7b8fm+JNBv\/ly1jJA3rlXBH\/oRr6K1bUzcz5aNvkHyqD93\/wCvXgf7X1tFceG7TUcbHgvQid9wdSCP\/HRUTWhUWfPM30GfpRSS9+aKxLP\/2Q==","photoid":"618fb893-cef1-47f2-9533-1fba4bb1b148","profileid":891,"screenname":"david5451"},"genderid":2,"hasgalleryphoto":true,"id":891,"interestdate":null,"lastloggedonstring":"Over one Month","lastlogindate":"\/Date(1327192022020-0600)\/","latitude":31.896944046020508,"likedate":null,"longitude":34.818611145019531,"lookingForageto":null,"lookingforagefrom":null,"lookingforid":null,"mycatchyintroline":null,"online":false,"peekdate":null,"postalcode":"","postalcodestatus":false,"profilephotos":null,"screenname":"david5451","stateprovince":""}]