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

**Descritipon :**

Handles the most basic member actions such as retrieving profileids etc

**Help URL :**

<http://173.160.122.195/Anewluv.Web.MediaService/PhotoService.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**

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

1) **addphotos**– allows an user with valid password and username and apikey in request header to be able to upload multiple photos converted into base64 string.

**Help url**

[http://173.160.122.195/Anewluv.Web.MediaService/PhotoService.svc/Rest/help/operations/addphotos](http://localhost/Anewluv.Web.MediaService/PhotoService.svc/Rest/help/operations/addphotos)

**url**

[http://173.160.122.195/Anewluv.Web.MediaService/PhotoService.svc/Rest/addphotos](http://localhost/Anewluv.Web.MediaService/PhotoService.svc/Rest/addphotos)

Json Body

{

"autoupload":true,

"multiple":true,

"photosuploaded":[{

"approvalstatusid":1,

"caption":"String content",

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

"imagebytes":null,

"imageb64string":" /9j/4AAQSkZJRgABAQEASABIAAD/2wBDAA0JCgsKCA0LCgsODg0PEyAVExISEyccHhcgLikxMC4pLSwzOko+MzZGNywtQFdBRkxOUlNSMj5aYVpQYEpRUk//2wBDAQ4ODhMREyYVFSZPNS01T09PT09PT09PT09PT09PT09PT09PT09PT09PT09PT09PT09PT09PT09PT09PT09PT0//wgARCAEgAYADASEAAhEBAxEB/8QAGgAAAwEBAQEAAAAAAAAAAAAAAQIDBAAFBv/EABgBAQEBAQEAAAAAAAAAAAAAAAEAAgME/9oADAMBAAIQAxAAAAHwzoBrmQhgCct3Vxn1sWamfVEpxZug9RNcaBZtRbjz1dM6WaVtzVSc1k+X7rg9XGjzBSNGHVxpmrVp568leuS1HkPUxoga2qA0X9jlus8M6sy82TtGjfKDrzYUTXLXCuduah1E0+”,

"imagename":"String content",

"imagetypeid":1,

"photostatusid":1,

"rejectionreasonid":null,

}],

"profileid":2

}

Notoable items

the paramters you need :

"**autoupload":true** - set this to true

"**multiple":true,** - true if we have multiple photos to upload false if only one

the generic list of objects will be the **photosuploaded** object

parameters :

**"approvalstatusid":null, -** this should be null for each photo

**caption":"String content**" - the caption of the photo i.e " me and my sister"

**creationdate":"\/Date(928167600000-0500)\/"** - optional date if you want to pass a specific date you can if you want but the default date is date of the server.

**imageb64string ":** <- you must locally convert the image into a byte array and pass that as string (base 64 string)

"**imagename":"teest", - image name just the name of the image i.e meandmysister.jpg**

**"imagetypeid":**1,

\*\*this is from a list value : usee the call to get the image types from this call : \*i

**"photostatusid":1,**

\*\*this is from a list value : usee the call to get the image types from this call : \*i

**"rejectionreasonid":null**, - set as null

i,e you could be uploading a bunch of photos at once

To get the imagetypeID and Phosotstatusid values call the lookup service and look at these two methods

For now just pass 1’s for both values

Result

The profileID is rejjturned as an ID i.e 45 etc.

2)**getprofileidbyscreenanme** – gets the profileID by screenname

**Help url**

[**http://173.160.122.195/Anewluv.Web.MembersService/MembersService.svc/Rest/help/operations/getprofileidbyscreenname**](http://localhost/Shell.MVC2.Web.MembersService/MembersService.svc/Rest/help/operations/getprofileidbyscreenname)

**url**

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

Json Body

{

**"username":"myusername"**

}

Username is required

Result

The profileID is returned as an ID i.e 45 etc.