**Basic Api Call documentation for NEW MemberActivity (Members area)**

**Created : 8-31-2013 ola lawal**

Important notes:

User profile must be first authenticated , and the profile id must be re-trived to use the services internal to Anewluv.

\*\* Coming soon : API key will be required on an app per app basis to access the member services. So as you code note you might have to add the follow header to your JSON calls \*\*

The Member Activity Page/Area

To access these API’s user must be validated via either OPEN ID ( face book, Yahoo,Google) or via the api with their username and password

**AuthenticationService.Docx** ( containes the documentation for required authentication calls)

List of API service calls within the authenciation service used to validate the user profile :

1. UserName and Password -

see AuthenticaionService APi document use the call : **validateuserbyusernamepassword**

1. OpenID/Email (i.e facebook , google , yahoo)

See AuthenticationServiceApi document use the call : **validateuserbyopenid**

**Main Service calls needed broken down by section**.

Steps -

Step 1) Authenticaiton

Once user is authenticated access to the MemberActivity page is allowed.

After the user is logged in either via registration APIs or the Login api the user’s profile ID should be cached or locally stored to be used with the API. To re-retrive the profile id see documentation in

MemberService.docx(containes documentation for the member service calls for getting individual profile data)

There are three ways to get the profile id similar to the authentication.

1. If the user name logged in with username/password use the following service call

See MemberService.docx use the call : **getprofileidbyusername**

1. If the user name logged in with Face book or open Id use the following service call

See MemberService.docx use the call : **getprofileidbyopenid**

1. If the user is already logged on or you do not have acces to either of those items above use

See MemberService.docx use the call : **getprofileidbyscreenname**

**Accessing Member Activity Links guidelines and Directions**

On the web Member activity should be accessible using URL (for SEO purposes) can be accessed directly

See the above section for mapping URL’s to the correct ativity type

1. /Members/MemberActivity/NewPeeks
2. /Members/MemberActivity/Peeks
3. /Members/MemberActivity/MyPeekes
4. /Members/MemberActivity/NewLikes
5. /Members/MemberActivity/Likes
6. /Members/MemberActivity/MyLikes
7. /Members/MemberActivity/NewInterests
8. /Members/MemberActivity/Interests
9. /Members/MemberActivity/MyInterests
10. /Members/MemberActivity/MyBlocks
11. /Members/MemberActivity/NewFriendRequests
12. /Members/MemberActivity/Friends
13. /Members/MemberActivity/FriendRequestsSent

**Data Retrevial** :

**Preleminary Data needed: ( the total count of the Activity type being viewed) PAGING and Virtual Scrolling requirement for this page.**

**This data should not be cached! It should be queried first using a callback and once the value from the callback is retrived it should then be used to for the call the retrieve the current page data.**

**Also as user’s page or scroll this data should be re-quried and counts updated in case data has changed during current user activity.**

Bedfore retrieving the data for a page the count should be retived first these counts are the same information retrived for display on the homepage. The counts are needed becase the data that populates the member activity is paged.

\*\* note the total number of values should be displayed on the page and paging an virtual scrolling is alis used.. scrolling is virtual

Virtual Scrolling : basically a scroll bar or an arrow at the If there are values above and bottom if there are values below.

WEB examples using KnockoutJS :

<http://www.dotnetcurry.com/ShowArticle.aspx?ID=911>

<http://smellegantcode.wordpress.com/tag/virtual-scrolling/>

Paging: Same paging as used on the the maches area

The counts come from the following service calls :

1. /getwhopeekedatmenewcount/{profileid}
2. /getwhoislikesmenewcount/{profileid}
3. /getwhoisinterestedinmenewcount/{profileid}

Here are the service calls for the activity datas again see above for what calls map to what activity type

Profile ID is self evident , Page is the current page of data to display, numberperpage is the number to display i.e 4 would mean 4 reccods would be returned in that case.

1. /getwhopeekedatmenew/{profileid}/{page}/{numberperpage}
2. /getwholikesmenew/{profileid}/{page}/{numberperpage}
3. /getwhoisinterestedinmenew/{profileid}/{page}/{numberperpage}

See the service call details are found in document for **MemberActionsService.docx**

**Paging Example**

I.e say a member has 20 total likes. On the member activity page the default view allows for 4 profiles displayed per page (WEB) less on mobile depending on design. So to retrieve the first page of data the call would

Profile ID = 1 , Number per page = 4 current page = 1

<http://173.160.122.195/MemberActionsService/MemberActionsService.svc/Rest/getwholikesme/1/1/4>

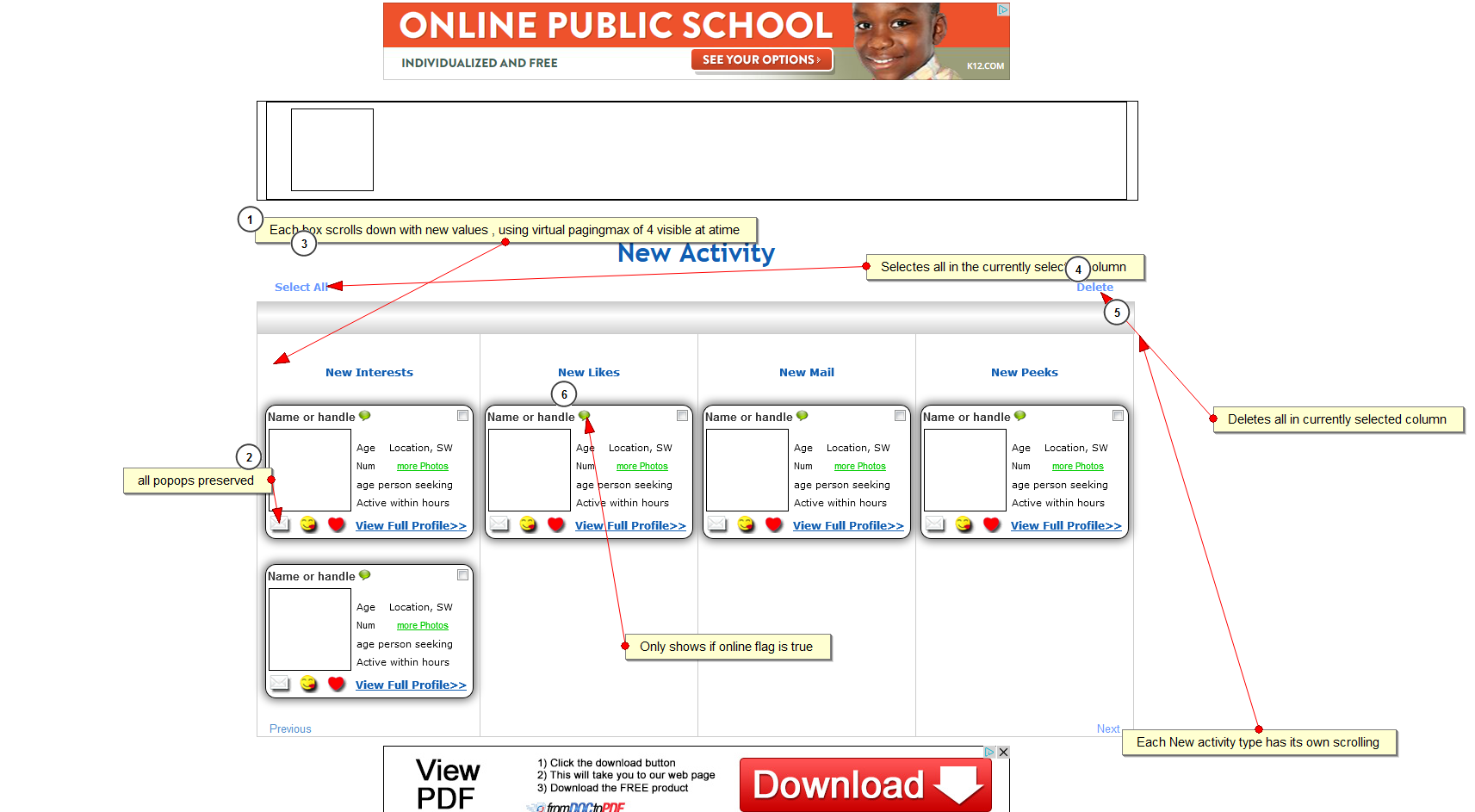
**\*\* for aditionaly documentation seee the service help page :**

[**http://173.160.122.195/.MemberActionsService/MemberActionsService.svc/Rest/help**](http://173.160.122.195/.MemberActionsService/MemberActionsService.svc/Rest/help)

You are now ready to make the service calls to populate the MemberActivity page using the following links from the home page

Step 2) Display Activity data based on activity type

This screen shows a summary



Coulums are populated similar to the the member activity documentation, the only diffrenece is all the service calls are made on this page.