

# LinkedDominator 2.0.0.8

## System Design Documentation 1.0.0.1

### **Document History:**

**Created By: Prabhat Sinha**

Revision Number	Revision Date	Version	Description
1	24-Dec-2014	1.0.0.1	This document has been cover all features and technical part of Linked Dominator.
2	26-Dec-2014	1.0.0.2	

### **Overview:**

This is a technical blueprint for the project. This document has been developed by Technical Department of Linked Dominator for the website <http://www.linkeddominator.com/>. LinkedIn provides numerous social media advertising solutions to increase brand awareness and lead generation. You can reach the most influential and educated audience with your well-managed LinkedIn marketing strategies.

1. Main Feature	Page#
1.1 BaseLib	02-03
1.2 Back End (Database)	03-04
1.3 Licensing.	04-07
1.4 Account Manager	07-08
1.5 Add Connection	08-15
1.6 LinkedIn Search	15-18
1.7 LinkedIn Scrapper	19-22

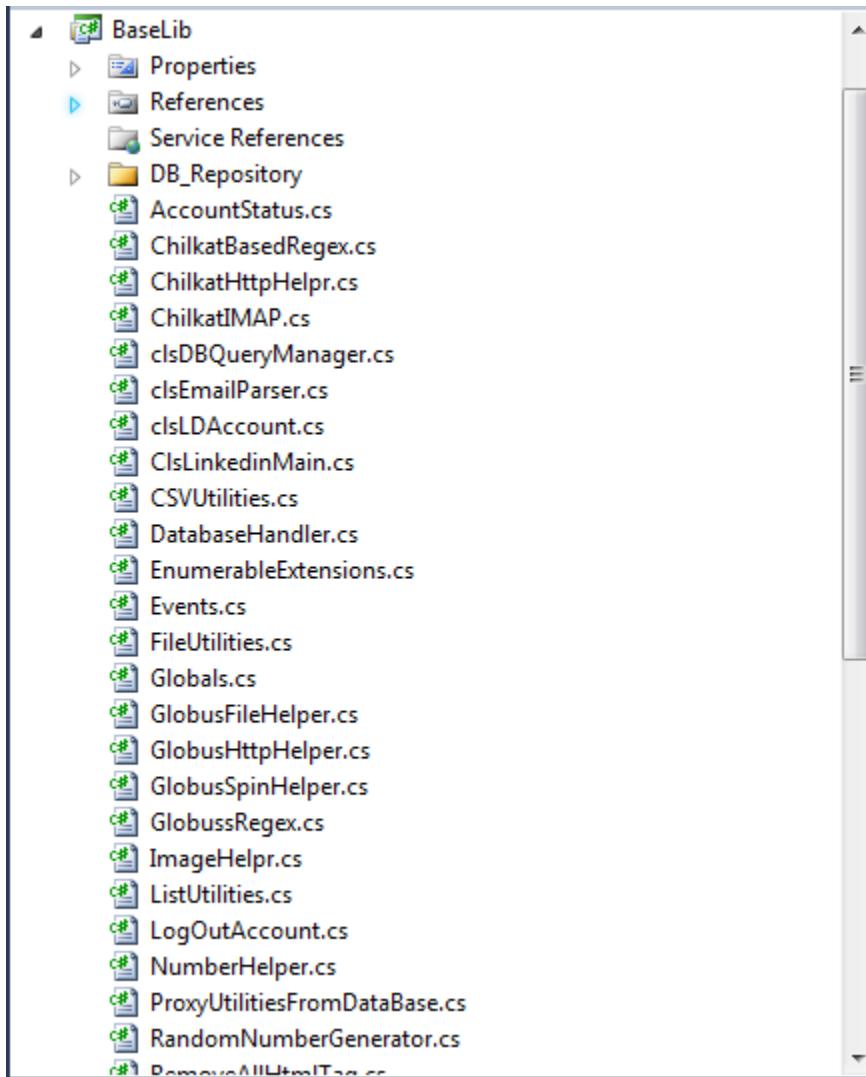
1.8 LinkedIn Campaign Scraper	22-13
1.9 Status Update	23-26
1.10 Create Group	26-30
1.11 Join Friends Group	30-35
1.12 Join Search Group	35-40
1.13 Group Status Update	40-43
1.14 Compose Message	43-48
1.15 Message Group Member	48-53
1.16 Proxy Setting	53-56
1.17 Endorse Your Profile	56-59
1.18 Profile Ranking	60-61
1.19 Share URL Link	62-63
<b>2 Other Features</b>	
2.1 Accept Invitation	64-66
2.2 Invite Member through Profile URL	66-68
2.3 Join Group using URL	68-70
2.4 Follow Company using URL	70-72
2.5 Remove Pending urls	72-75
2.6 Campaign Invite with Search	76-78
2.7 Invite Members to Groups	78-81

### **1.1 BaseLib**

BaseLib (Class Library) – It's a library which contains all the methods/functionalities which other projects in the solution would use.

Functionalities of BaseLib:-

- 1) Making all Http Web Requests like Get, Post, Multipart uploading. Managed cookies between requests.
- 2) File Handling methods like Reading/Writing into Text/CSV, Excel.
- 3) String operations, Spinning, Random number generations.
- 4) Events classes for managing events firing, and passing data through events



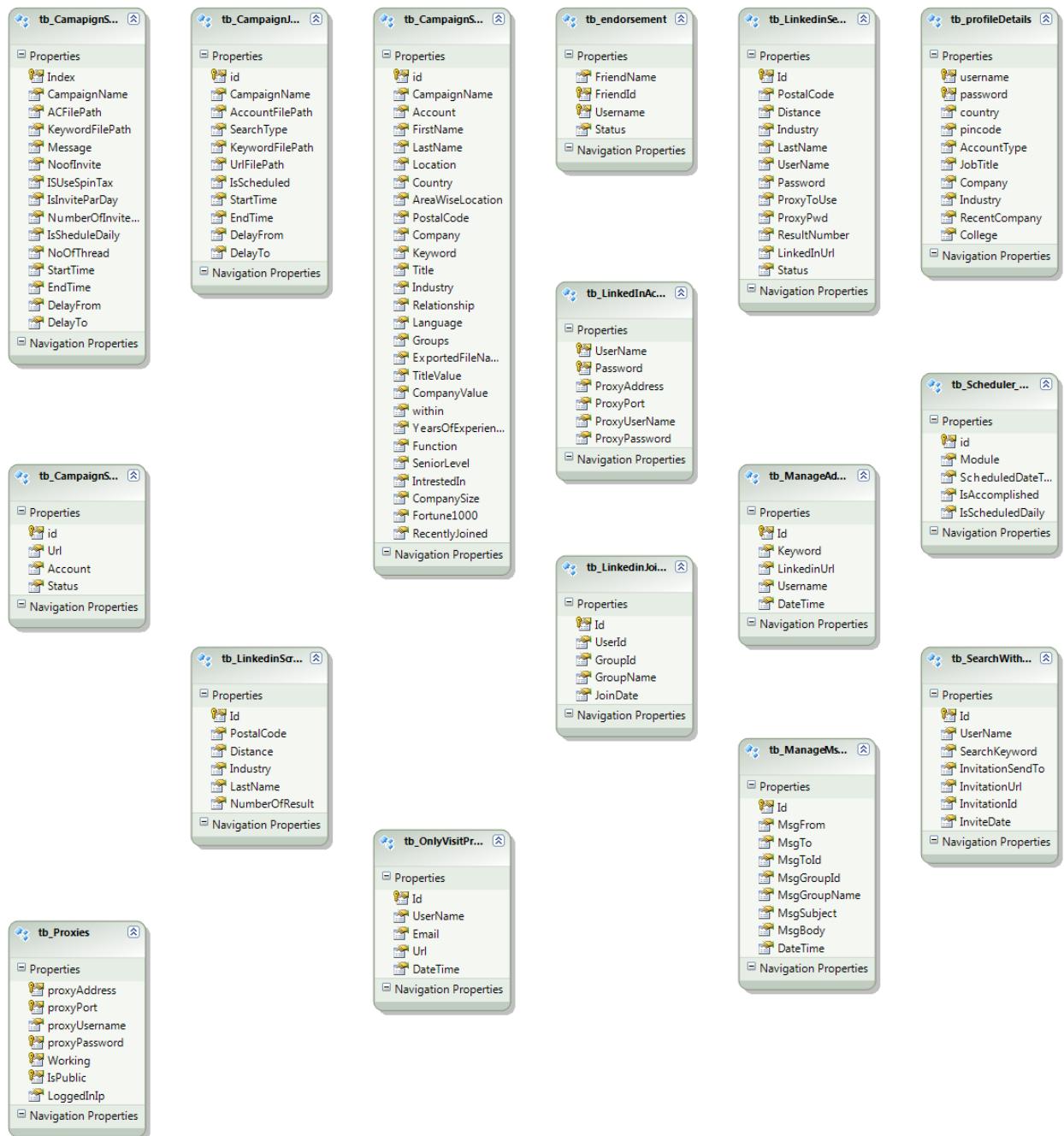
## 1.2 Backend

SQLite – Here we store all the relevant details for the LinkedIn accounts used by the application user, campaigns, posts, and many more. We store this database locally in App Data folder.

Path: C:\Users\GLB-110\AppData\Local



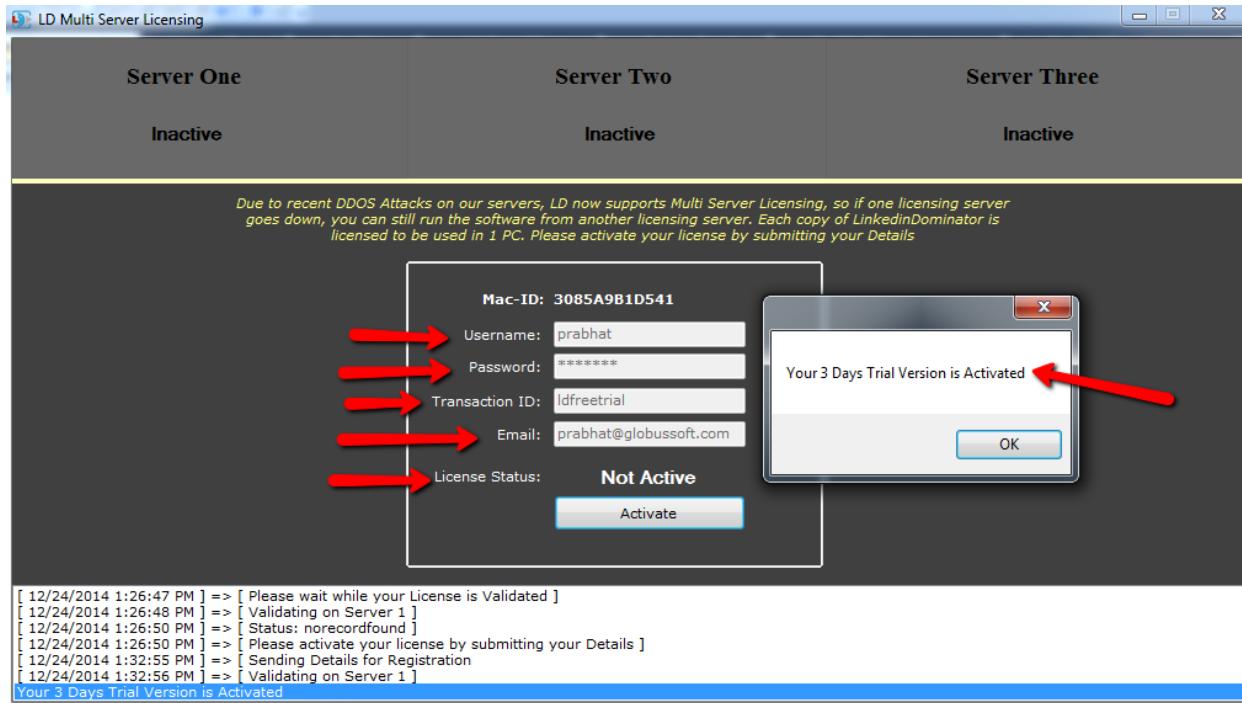
## LD DB Entity Design:



### 1.3 Licensing.

If start the LD project licensing module asking for fill detail which is basically two option (i) freetrial (ii) one month or Lifetime licensing.

- (i) Freetrial is 3days trial option if fill details like,Username,password,TransactionID,email MACID automatic detected by every single system. After 3 days its automatic expire.



After submit Activate key its check the License validation in method:

StartLicenseValidation()

Its pass all parameters shown below,

(licensemanager.ValidateCPUID(**ref** status, server1, **ref** username, **ref** pass, **ref** txnID, **ref** email, freeTrialKey, cpuID))

And check its freetrial or other with GetRequest

GetRequest,

Servr = "[linkeddominator.com/licensing/LD](http://linkeddominator.com/licensing/LD)" Its pointing with LinedDominator.com site

```
res = HttpHelpR.GetHtml("http://" + servr + "/GetUserData.php?cpid=" + cpuID + "");
```

After get request its get the status if its return "no record found"  
Its insert the all details fill by user.

Remote site: /licensing

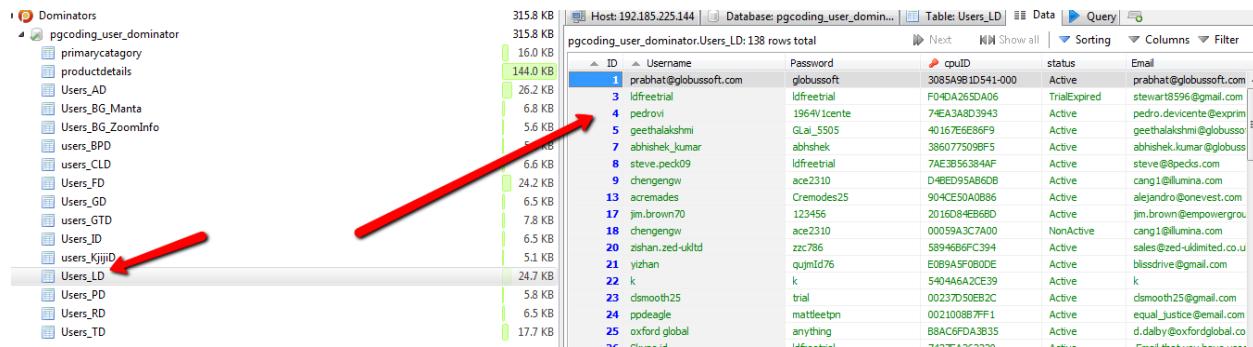
Filename	Filesize	Filetype	Last modified	Permissions
..				32
ID		File folder	12/5/2014 5:31:...	0755
LD		File folder	11/14/2014 9:5:...	0755
LDLatestVersion.txt	88	Text Docu...	12/10/2014 5:1...	0644
LDLatestVersion_64.txt	90	Text Docu...	12/10/2014 6:4...	0644
Setup_LinkedDominator_32_10.12.14_2.8.zip	7,012,513	WinRAR ZI...	12/10/2014 5:0...	0644
Setup_LinkedDominator_64_10.12.14_2.8.zip	195,736	WinRAR ZI...	12/10/2014 6:5...	0644

All necessary PHP scripts for licensing which is responsible for LD Licensing,

Remote site: /licensing/LD

Filename	Filesize	Filetype	Last modified	Permissions
..				32
class		File folder	11/13/2014 3:3...	0755
checkLicence.php	353	PHP File	11/13/2014 3:4...	0755
datetime.php	34	PHP File	11/13/2014 3:3...	0755
GetUserData.php	1,059	PHP File	11/13/2014 4:1...	0755
register.php	980	PHP File	11/13/2014 4:1...	0755
UpdateStatus.php	340	PHP File	11/17/2014 7:1...	0755

All licensing details have maintained separate Mysql db see below,



Host: 192.185.225.144 | Database: pgcoding\_user\_domin... | Table: Users\_LD | Data | Query | [Next](#) | [Show all](#) | [Sorting](#) | [Columns](#) | [Filter](#)

ID	Username	Password	cpID	status	Email
1	prabhat@globussoft.com	globussoft	3085A9B1D541-000	Active	prabhat@globussoft.com
3	ldfreetrial	ldfreetrial	F04DA265DA06	TrialExpired	stewart8596@gmail.com
4	pedrovi	1964Vicente	74E4A3AB03943	Active	pedro.devicente@exprim...
5	geethalakshmi	Ghai_5505	40167E6E06F9	Active	geethalakshmi@globusso...
7	abhishek_kumar	abhishek	386077509BF5	Active	abhishek.kumar@globus...
8	steve.ped09	ldfreetrial	7AE3B56384AF	Active	steve@specks.com
9	chengengw	ace2310	D4BED95A80DB	Active	cang1@lumina.com
13	acremades	Cremodes25	904CE50A0896	Active	alejandro@onevest.com
17	jim.brown70	123456	2016084EB6BD	Active	jim.brown@empowergrou...
18	chengengw	ace2310	00059A3C7A00	NonActive	cang1@lumina.com
20	zishan.zed-ultd	zzc786	5894686FC394	Active	sales@zed-ultdlimited.co.u...
21	yizhan	qjymId76	E089A5F0800E	Active	blissdrive@gmail.com
22	k	k	540446A2CE39	Active	k
23	dsmooth25	trial	00237D50EB2C	Active	dsmooth25@gmail.com
24	ppdeagle	mattieeton	00210087FF1	Active	equal_justice@mail.com
25	oxford global	anything	B8AC6FDA3B35	Active	d.daby@oxfordglobal.co...
26	elton_wd	146666666666	7A07EA0A0000	Active	elton_wd@...

If free trial its Automatic activated but if not a free trial means if client are purchase the software it's not activated automatically support guy have activated the license after check the PayPal status.

#### 1.4 Account Manager

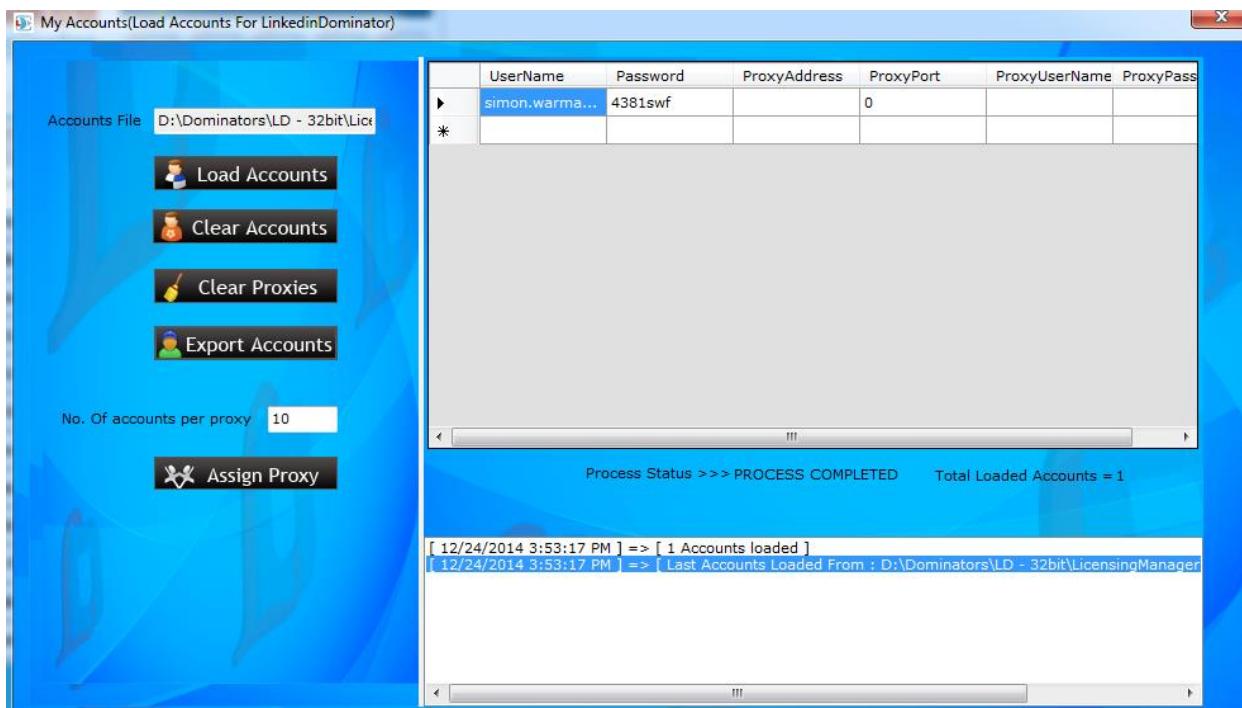
Account master is very important role for LinkedDominator its master record of all accounts manageable by LinkedIn features. Accounts has been save in DB\_LinkedInDominator db.

Without filling account here not working any features of LinkedInDominator.

SQLite – Here we store all the relevant details for the LinkedIn accounts used by the application user.

Db Path: C:\Users\GLB-110\AppData\Local\LinkedInDominator

 DB_LinkedInDominator	12/24/2014 3:53 PM	Data Base File	5,018 KB
--	--------------------	----------------	----------



## Features,

### a). Load Account:

If you have add some new accounts click Load account its option to select your account with: "username:password" format also added accounts with proxies both (Public.Private) like "username:password:proxyaddress:port:proxypassword"

How it work with code,

mainclass "clsSettingDB" is responsible for all functionality of Account manager like, insert,update and delete the records used by same module.

```
clsSettingDB ObjclsSettingDB = new clsSettingDB();
ObjclsSettingDB.InsertOrUpdateSetting("LoadAccounts", "LoadAccounts",
StringEncoderDecoder.Encode(txtAccountFile.Text));
```

### b). Clear Account:

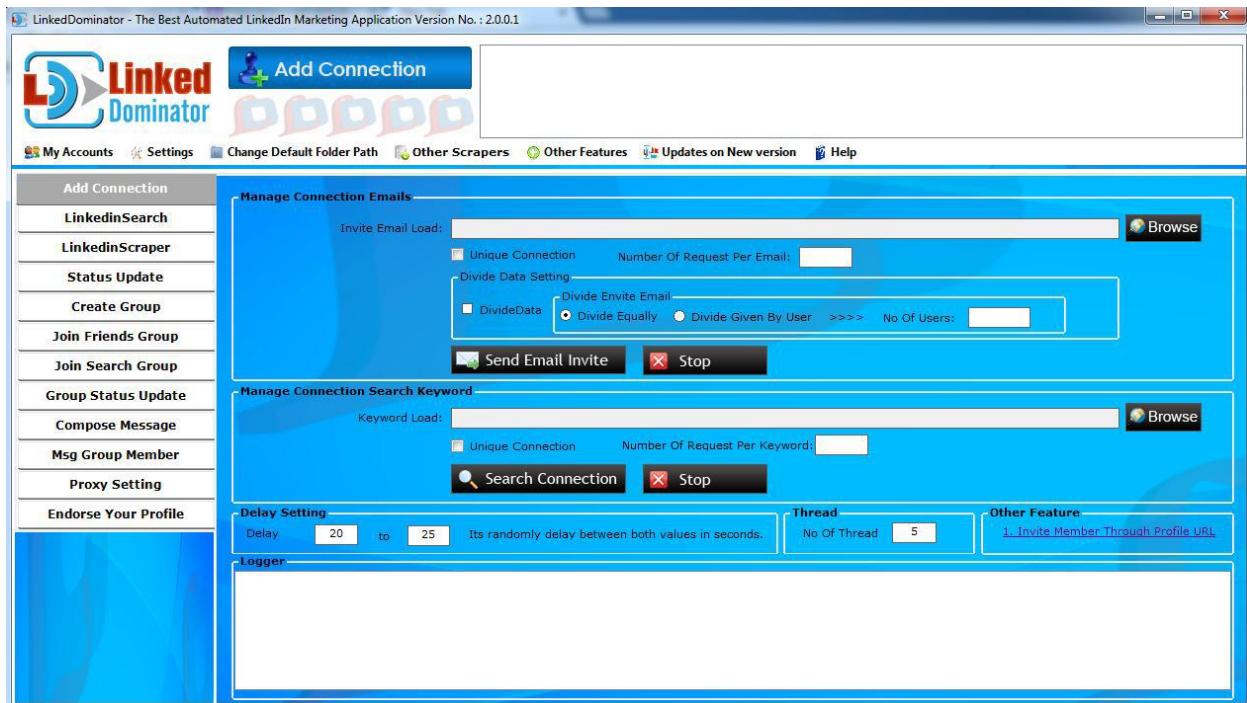
```
DataBaseHandler.DeleteQuery(DeleteQuery, "tb_LinkedInAccount");
```

### b). Clear Proxy:

Clear all proxy which is added in accounts.

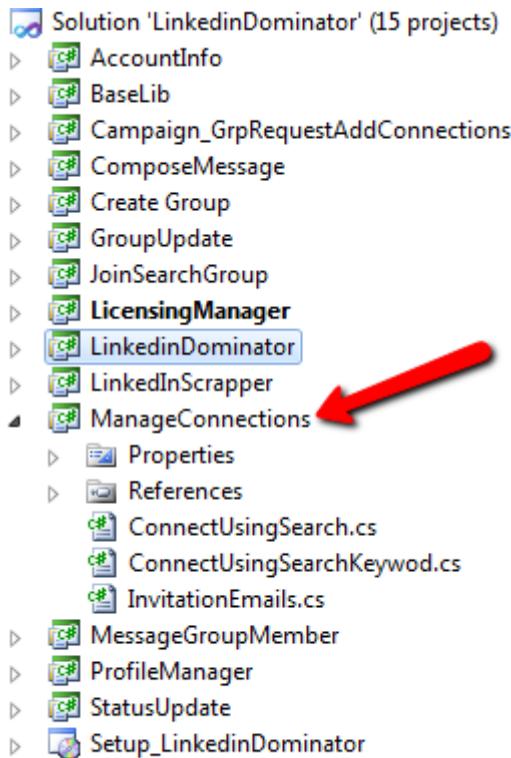
## 1.5 Add Connection

if you want to send connection request to certain people (related to any keyword) or you have bunch of emails to whom you want to add in your connection then you can have this feature and send connection request to people in both ways.



- If you have a list of emails then write them down in a text file and upload them in a 'upload email load' section.

Separate class are maintain for Add connection Module,



### **Manage connection with Emails:**

How it work with code,

If you have click browse button with email keyword, option to choose email in text format,

```

foreach (string item in templist)
{
  if (!string.IsNullOrEmpty(item.Replace(" ", "").Replace("\t", "")))
  {
    _lstInviteEmailConnection.Add(item);
  }
}
  
```

Options available like, unique connection, Number of request per email and divide data setting,db

```

public class ConnectUsing_Search
"ConnectUsing_Search"
  
```

Class are responsible for basic functionality of Add connection with email,



LinkedDominator Document Ver 1.0

After submit process start with single thread,

```
new Thread(() => InviteEmailConnectionThread()).Start();
```

all settings like thread setting and delay setting, divide data setting are initialize with method,

"**InviteEmailConnectionThread()**"

In this method its run with another method with multithreaded environment

```
ThreadPool.QueueUserWorkItem(new WaitCallback(SendInviteUsingEmails), new object[] {  
    item, lstInviteEmailConnection_temp, SearchMinDelay, SearchMaxDelay});  
index++;  
Thread.Sleep(1000);  
}
```

After current process its call another method,

All emails loaded by user are enqueue and dequeue with foreach loop in method,

"**SendInviteUsingEmails**"

After this process call another method "**InviteEmailConnect**"

```
LinkedIn_Master.InviteEmailConnect(ref invitationEmails,  
SearchCriteria.NumberOfRequestPerEmail, Email.MaximumDelay, Email.MinimumDelay);
```

LinkedIn\_Master

```
Invitation_Emails.InviteFriendThroughEmail(noofemailRequest, MaxDelay, MinDelay);
```

And class "**InviteFriendThroughEmail**" are responsible for final processing of this feature.

1. Account Login

```
linkedinLoginAndLogout.LoginHttpHelper(_UserName, _Password, _ProxyAddress,  
_ProxyPort, _ProxyUserName, _ProxyPassword, ref HttpHelper, ref xMESSAGE))
```

2. get request

```
PgSrcInviteMessgae = HttpHelper.getHtmlFromUrlProxy(new  
Uri("https://www.linkedin.com/fetch/importAndInviteEntry"), _ProxyAddress,  
proxyport, _ProxyUserName, _ProxyPassword);
```

3. Post Request

```
//Post Data for Invite Message  
PostData = "emailAddresses=" + Uri.EscapeDataString(emailAddresses) +  
"&subject=Invitation+to+connect+on+LinkedIn&csrfToken=" + csrfToken +  
"&sourceAlias=" + sourceAlias;
```

```
//Post Url for Invite Message  
postUrl = "https://www.linkedin.com/fetch/manual-invite-create";  
string postResponse = HttpHelper.postFormDataRef(new Uri(postUrl), PostData,  
"https://www.linkedin.com/fetch/manual-invite-create", "", "", "", "", "");
```

### **Manage connection with Keyword:**

How it work with code,

If you have click browse button with email keyword, option to choose email in text format,

```
foreach (string item in templist)
{
    if (!_lstConnectionSearchKeyword.Contains(item))
    {
        if (!string.IsNullOrEmpty(item.Replace(" ", "").Replace("\t", "")))
        {
            _lstConnectionSearchKeyword.Add(item);
        }
    }
}
```

Options available like, unique connection, Number of request per keyword, only visit profile and clear database etc.

Class “[ConnectUsing\\_Search](#)” are responsible for basic functionality of Add connection with keyword, After submit process start with single thread,

```
if ((LinkedInManager.linkedInDictionary.Count > 0 && _lstConnectionSearchKeyword.Count > 0 && (!string.IsNullOrEmpty(txtNumberOfRequestPerKeyword.Text)))
{
    _GroupMemberSearch = LinkedInManager.linkedInDictionary.Count;
    btnSearchConnection.Cursor = Cursors.AppStarting;

    new Thread(() => SearchConnectionKeywordThread()).Start();
}
```

After all setting like thread setting, delay setting its call another method with multithreaded environment

```
foreach (KeyValuePair<string, LinkedInMaster> item in
LinkedInManager.linkedInDictionary)
{
    ThreadPool.SetMaxThreads(SetThread, 5);
    ThreadPool.QueueUserWorkItem(new WaitCallback(SendInviteUsingKeyWords), new
object[] { item, SearchMinDelay, SearchMaxDelay });
}
```

After this process its call another method,  
All keyword loaded by user are enqueue and dequeue with foreach loop in method,

“SendInviteUsingKeyWords”

After this process call another method “SendInviteUsingKeyWords”

```
LinkedIn_Master.ConnectSearchUsingkeyword(ref ConnectUsing_Search, SearchMinDelay,  
SearchMaxDelay);
```

After call another class “ConnectUsing\_Search”

In method,

```
ConnectUsing_Search.ConnectionSearch(SearchMinDelay, SearchMaxDelay);
```

class “ConnectionUsingSearch” are responsible for final processing of this feature.

In method ConnectionSearch

1. Account Login

```
linkedinLoginAndLogout.LoginHttpHelper(_UserName, _Password, _ProxyAddress,  
_ProxyPort, _ProxyUserName, _ProxyPassword, ref HttpHelper, ref Textmessage)
```

2. First Get Request

```
string Url = "http://www.linkedin.com/search/fpsearch?keywords=" +  
_ConnectSearchKeyword +  
"&keepFacets=keepFacets&page_num=1&pplSearchOrigin=ADVS&viewCriteria=2&so  
rtCriteria=R&redir=redir";
```

```
string PgSrcMain = HttpHelper.getHtmlfromUrlProxy(new Uri(Url), _ProxyAddress,  
proxyport, _ProxyUserName, _ProxyPassword);
```

3. Second Get Request

```
string urlGetdata = "http://www.linkedin.com/vsearch/f?keywords=" +  
_ConnectSearchKeyword +  
"&orig=GLHD&pageKey=voltron_federated_search_internal_jsp&search=Search";  
urlGetdata =  
"http://www.linkedin.com/vsearch/f?keywords=Jobs&orig=GLHD&pageKey=voltron_  
federated_search_internal_jsp&search=Search";  
PgSrcMain = HttpHelper.getHtmlfromUrlProxy(new Uri(Url), _ProxyAddress, proxyport,  
_ProxyUserName, _ProxyPassword);
```

After getting page no's its call another methods with for loop according to pagination

If setting with unique connection call,

```
TestConnectPageSearchForUniqueUrl(i.ToString(), SearchMinDelay, SearchMaxDelay);
```

Without unique connection,

```
TestConnectPageSearch(i.ToString(), SearchMinDelay, SearchMaxDelay);
```

```

for (int i = 1; i <= PageNumber; i++)
{
    //loop for send request
    if (!Isaccountvalid)
    {
        return;
    }

    if (!UseuniqueConn)
    {
        TestConnectPageSearch(i.ToString(), SearchMinDelay, SearchMaxDelay);
    }
    else
    {
        TestConnectPageSearchForUniqueUrl(i.ToString(), SearchMinDelay, SearchMaxDelay);
    }
}

```

In method "TestConnectPageSearch"

1. 1<sup>st</sup> Get request

```

string UrlConnectpage = "http://www.linkedin.com/search/fpsearch?keywords=" +
    _ConnectSearchKeyword + "&keepFacets=keepFacets&page_num=" + pageNumber
    + "&pplSearchOrigin=ADVS&viewCriteria=2&sortCriteria=R&redir=redir";

string PgSrcMain1 = HttpHelper.getHtmlfromUrlProxy(new Uri(UrlConnectpage),
    _ProxyAddress, proxyport, _ProxyUserName, _ProxyPassword);

```

2. 2<sup>nd</sup> Get request

```

string thiredResponce = "http://www.linkedin.com/people/invite?from=profile&key=" +
    Val_key + "&firstName=" + Val(firstName) + "&lastName=" + Val(lastName) +
    "&authToken=" + Val(AuthToken) + "&authType=" + Val(AuthType) + "&csrfToken=" +
    Val(CsrToken) + "&goback=" + Val(goback);
string pageResponce2 = HttpHelper.getHtmlfromUrl1(new Uri(thiredResponce));

```

4. Date saved with SQLite table if setting "only visit profile"

```

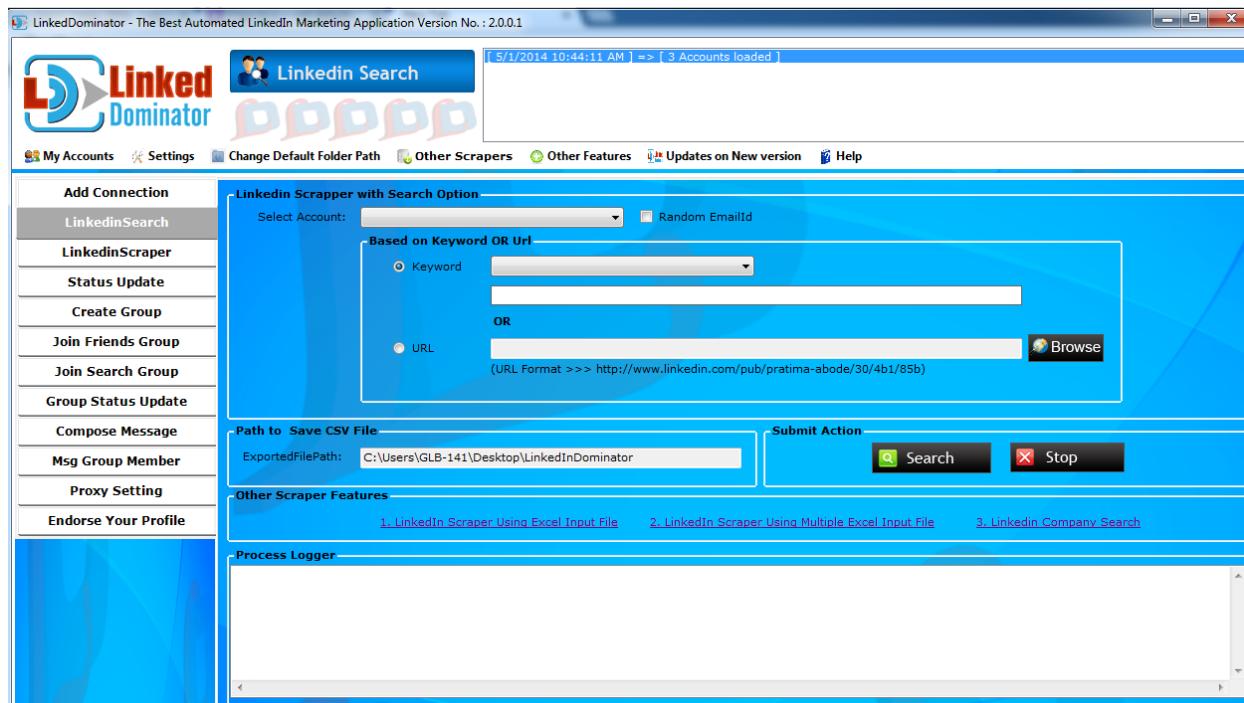
DataSet ds = new DataSet();
try
{
    string Querystring = "INSERT INTO tb_OnlyVisitProfile (Email,Url,DateTime) Values ('" +
    _UserName + "','" + url[0] + "','" + DateTime.Now + "')";
    ds = DataBaseHandler.SelectQuery(Querystring, "tb_ManageAddConnection");
}
catch { }

```

In "TestConnectPageSearchForUniqueUrl" method same functionality but only difference its work with unique connection.

## 1.6 LinkedIn Search

- a) Before starting the process please select the account in the dropdown from you which you want to run the process.
- b) The search is based on two things that are based on keyword and other one is based on company.

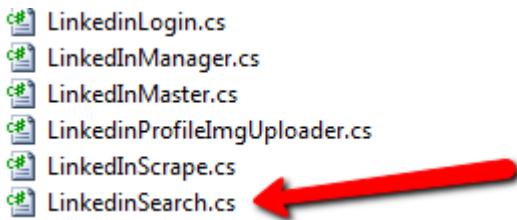


There are two option in LinkedIn Search,  
1t keyword wise scrape the data.  
2<sup>nd</sup> Url wise scrape the data.

There are two selection option in keyword, 1. People 2.Companies.

If you settings Keyword wise or Url wise scrape chose any one setting one time, after that click search button in "Submit Action" Section,

All basic methods of this module is written in LinkedIn Search class. See below,



How it work with code,

First execute method with single thread,

```
Thread thread_LinkedInSearch = new Thread(StartLinkedInSearch);
thread_LinkedInSearch.Start();
```

Thread execute with method “**StartLinkedInSearch**”

This method is responsible for Login,  
`LinkedInLogin` Login = `new LinkedInLogin()`;  
`Login.LogoutHttpHelper();`

After Login its call another method with Login reference,  
Method “**StartLinkedInSearch**” call with object

```
LinkedInDominator.LinkedInSearch obj_LinkedInSearch = new
LinkedInDominator.LinkedInSearch(item._Username, item._Password, item._ProxyAddress,
item._ProxyPort, item._ProxyUsername, item._ProxyPassword);
```

```
obj_LinkedInSearch.StartLinkedInSearch(ref HttpHelper);
```

In “**StartLinkedInSearch**” method is responsible with final process,

```
if (_RdbKeyword)
{
    if (_Search.Contains("People"))
    {
        try
        {
            SearchByPeople(ref HttpHelper);
        }
        catch{ }

    }

    if (_Search.Contains("Companies"))
    {
        try
        {
            SearchByCompany(ref HttpHelper);
        }
        catch{ }
    }
}
```

```
    }  
  
}
```

Search by people method,

### 1<sup>st</sup> get Request

```
pageSourceaAdvanceSearch = HttpHelper.getHtmlfromUrl1(new  
Uri("https://www.linkedin.com/vsearch/f?adv=true&trk=federated_advs"));
```

after this request will get csrfToken,

### 2<sup>nd</sup> get request

```
ResponseWallPostForPremiumAcc = HttpHelper.getHtmlfromUrl1(new  
Uri("http://www.linkedin.com/search/fpsearch?type=people&keywords=" + _Keyword +  
&pplSearchOrigin=GLHD&pageKey=fps_results"));
```

After Getting Total no of pages,

There are three types of post request in getting scrapped details,

```
for (int i = 1; i <= pagenumber; i++)  
{
```

//1. Account Type: Basic

```
PostRequestURL = "http://www.linkedin.com/search/fpsearch?";  
PostdataForPagination = "type=people&keywords=" + _Keyword + "&page_num=" + i;  
PostResponce = HttpHelper.postFormData(new  
Uri(PostRequestURL), PostdataForPagination);
```

//2. Account Type: Executive

```
PostRequestURL = "http://www.linkedin.com/search/hits";  
PostdataForPagination = "type=people&keywords=" + _Keyword + "&page_num=" + i;  
PostResponce = HttpHelper.postFormData(new Uri(PostRequestURL),  
PostdataForPagination);
```

//3. Account Type: Premium

```
PostRequestURL = "http://www.linkedin.com/search/fpsearch?";  
PostdataForPagination = "type=people&keywords=" + _Keyword + "&page_num=" + i;  
PostResponce = HttpHelper.postFormData(new Uri(PostRequestURL),  
PostdataForPagination);
```

```
}
```

After that get all Urls of scrapped data,

```
List<string> PageSerchUrl = ChilkatBasedRegex.GettingAllUrls(PostResponce,  
"profile/view?id");
```

Add all urls in liststring

```
RecordURL.Add(urlSerch);  
RecordURL.Distinct();
```

After collecting all urls start running final scrapping,

```
foreach (string url_item in lstRecordURLs)  
{  
    try  
    {  
        Log("[ " + DateTime.Now + " ] => [ Fetching Data From URL : " + url_item + " ]);  
  
        if (!CrawlingLinkedInPage(url_item, ref HttpHelper, status))  
        {  
            CrawlingPageDataSource(url_item, ref HttpHelper, status);  
        }  
    }  
    Catch {}  
}
```

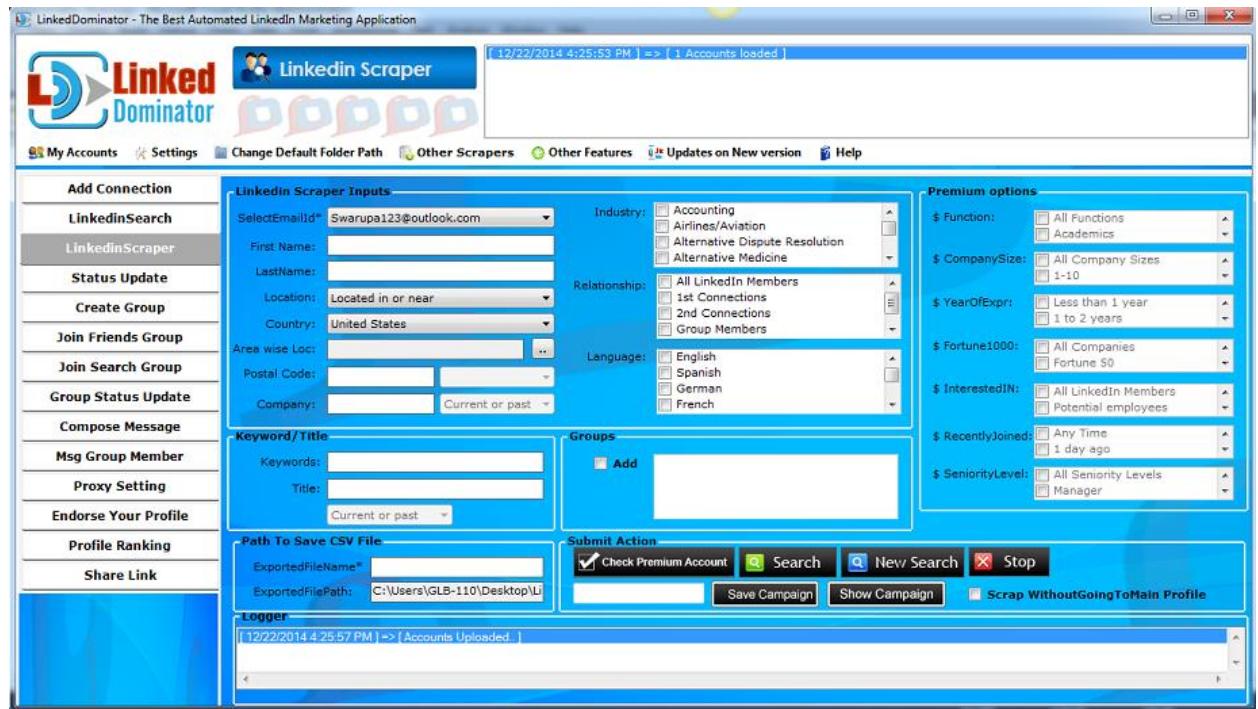
And finally scrape the data,  
Csv writing is responsible for class "**CSVUtilities**"

```
CSVUtilities.ExportDataCSVFile(CSVHeader,  
CSV_Content,Globals.path_LinkedinSearchByProfileURL);
```

Same Logic apply with scrape with URI.

### 1.7 LinkedIn Scraper

This is one of the powerful feature in the market for scraping profiles in LinkedIn. It has all those filters that LinkedIn has in their advance search section.



a) This is how the famous scraper looks like.

b) The process is just similar like the LinkedIn Search module. The good thing here is you don't need to upload any text file.

c) Just write down the data in the fields provided and the put one name for the out put file in the "ExportedFileName" field.

d) Then you will get the exported file.

#### **How it work with code,**

There are two types of accounts can scrape in LinkedInScrapper,

1. Basic Account
2. Premium Account.

In basic account some Limited search options  
 like, FirstName, LastName, Location, Country, Areawise  
 Locaption, postalCode, Company, Keyword, Title, Industry, Relaetionship, Langauge

Extra search option only for PremiumAccount, Function, Company Size, YearofExperince, Furtune1000, InterestedIn, RecentlyJoined,SeniorityLevel etc.

After setting selection option press "Search" Button,

Starting code with method "**StartLinkedInScrapper**" in single Thread,

```
new Thread(() =>
{
    StartLinkedInScrapper();
}).Start();
```

In "startLinkedInScrapper()"

Runnin login Process with,

```
HttpHelper = new GlobusHttpHelper();
LinkedInScrape objlinkscr = new LinkedInScrape();
bool isLoggedIn = Login_LinkedInScrapper();
```

After successfully login call another method "StartLinkedinScrapWithPagination",

```
if (isLoggedIn)
{
    AddLoggerScrapeUsers("[ " + DateTime.Now + " ] => [ Start LinkedIn Crawling ]");
    objlinkscr.StartLinkedinScrapWithPagination(ref HttpHelper);
}
```

In "objlinkscr.StartLinkedinScrapWithPagination"

After Getting Total no of pages,

There are three types of post request in getting scrapped details,

```
for (int i = 1; i <= pagenumber; i++)
{
    //1. Account Type: Basic

    PostRequestURL = "http://www.linkedin.com/search/fpsearch?";
    PostdataForPagination = "keywords=" + Uri.EscapeDataString(SearchCriteria.Keyword) +
    "&title=" + Uri.EscapeDataString(SearchCriteria.Title) + "&fname=" +
    SearchCriteria.FirstName + "&lname=" + SearchCriteria.LastName +
    "&searchLocationType=" + SearchCriteria.Location + "&f_FG=" + SearchCriteria.Group +
    "&companyScope=" + SearchCriteria.CompanyValue + "&countryCode=" +
    SearchCriteria.Country + "&company=" + SearchCriteria.Company +
    "&viewCriteria=1&sortCriteria=R&facetsOrder=CC%2CN%2CG%2CI%2CPC%2CED%2CL%2
```

```

CFG%2CTE%2CFA%2CSE%2CP%2CCS%2CF%2CDR&page_num=" + i +
"&openFacets=N%2CCC%2CG";
PostResponce = HttpHelper.postFormData(new
Uri(PostRequestURL),PostdataForPagination);

//2. Account Type: Executive
PostRequestURL = "http://www.linkedin.com/search/hits";
PostdataForPagination = "keywords=" + Uri.EscapeDataString(SearchCriteria.Keyword) +
"&title=" + Uri.EscapeDataString(SearchCriteria.Title) + "&fname=" +
SearchCriteria.FirstName + "&lname=" + SearchCriteria.LastName +
"&searchLocationType=" + SearchCriteria.Location + "&f_FG=" + SearchCriteria.Group +
"&companyScope=" + SearchCriteria.CompanyValue + "&countryCode=" +
SearchCriteria.Country + "&company=" + SearchCriteria.Company +
"&viewCriteria=1&sortCriteria=R&facetsOrder=CC%2CN%2CG%2CI%2CPC%2CED%2CL%2
CFG%2CTE%2CFA%2CSE%2CP%2CCS%2CF%2CDR&page_num=" + i +
"&openFacets=N%2CCC%2CG";
PostResponce = HttpHelper.postFormData(new Uri(PostRequestURL),
PostdataForPagination);

//3. Account Type: Premium
PostRequestURL = "http://www.linkedin.com/search/fpsearch?";
PostdataForPagination = "keywords=" + Uri.EscapeDataString(SearchCriteria.Keyword) +
"&title=" + Uri.EscapeDataString(SearchCriteria.Title) + "&fname=" +
SearchCriteria.FirstName + "&lname=" + SearchCriteria.LastName +
"&searchLocationType=" + SearchCriteria.Location + "&f_FG=" + SearchCriteria.Group +
"&companyScope=" + SearchCriteria.CompanyValue + "&countryCode=" +
SearchCriteria.Country + "&company=" + SearchCriteria.Company +
"&viewCriteria=1&sortCriteria=R&facetsOrder=CC%2CN%2CG%2CI%2CPC%2CED%2CL%2
CFG%2CTE%2CFA%2CSE%2CP%2CCS%2CF%2CDR&page_num=" + i +
"&openFacets=N%2CCC%2CG";
PostResponce = HttpHelper.postFormData(new Uri(PostRequestURL),
PostdataForPagination);

}

```

Collecting all url with below code,

```

RecordURL.Add(urlSerch);
if (!queRecordUrl.Contains(urlSerch))
{
    queRecordUrl.Enqueue(urlSerch);
}
RecordURL = RecordURL.Distinct().ToList();

```

After collecting all urls again its call another thread with method “`finalUrlCollection`”,

```
new Thread(() =>
{
    if (SearchCriteria.starter)
    {
        string CheckString = string.Empty;
        finalUrlCollection(CheckString);

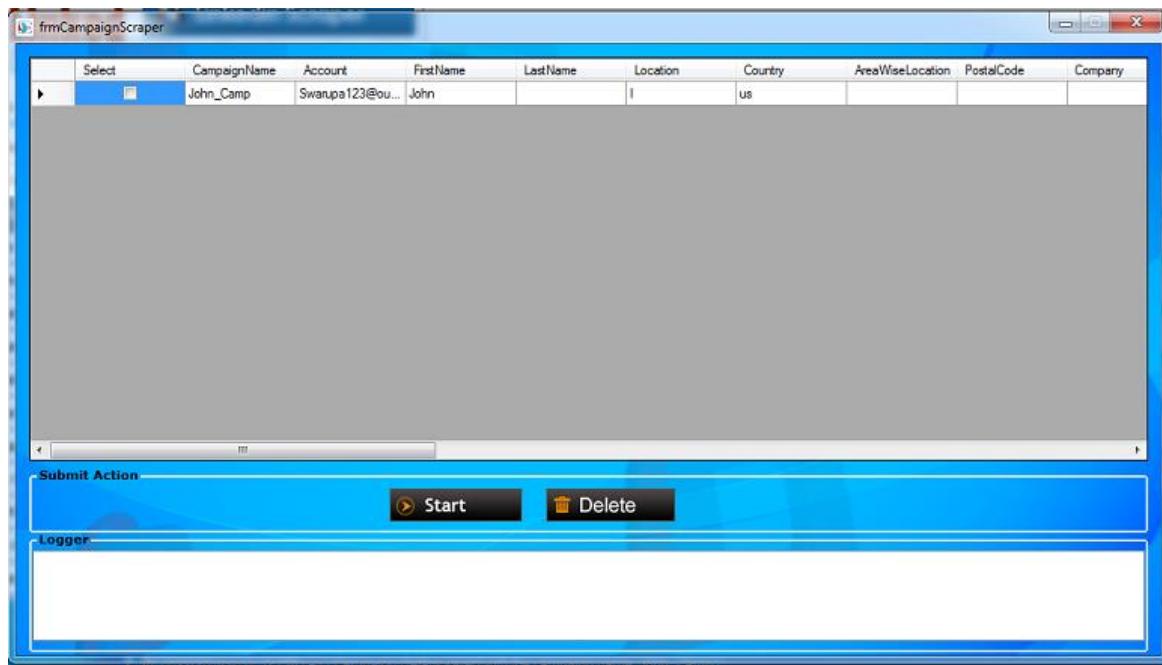
    }
}).Start();
```

And final process scrapping with csv generation with method “`CrawlingLinkedInPage`”

```
bool check = CrawlingLinkedInPage(urltemp, ref HttpHelper);
```

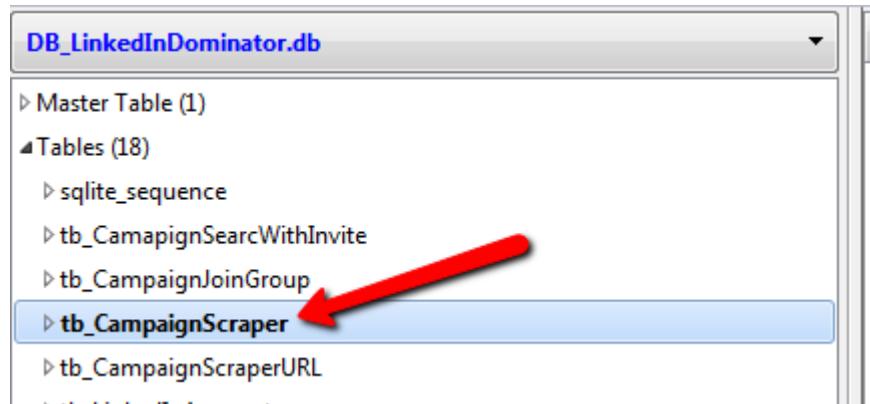
### **1.7.1 LinkedIn Campaign Scraper**

This is one of the powerful feature in the market for scraping profiles in LinkedIn. It has all those filters that LinkedIn has in their advance search section and save with campaign more than one campaign run at a time, same time we have created one csv for each campaign



#### **How it work with code,**

Select campaign which have generated by LinkedIn Scrapper Main page and save all campaigns in particular SQLite db,



It's run with multithreaded mode which is selected campaigns by user,  
All parameters passes with object model,

```
ThreadPool.QueueUserWorkItem(new WaitCallback(StartProcessWithMultiThread), new
object[] { campname, AccountData, FirstName, LastName, Location, Country, LocationArea,
PostalCode, Company, Keyword, Title, IndustryType, Relationship, language, Groups,
FileName, TitleValue, CompanyValue, within, YearsOfExperience, Function, SeniorLevel,
IntrestedIn, CompanySize, Fortune1000, RecentlyJoined });
```

If run “[StartProcessWithMultiThread](#)” method its call another method,  
“[StartLinkedInScraper](#)”

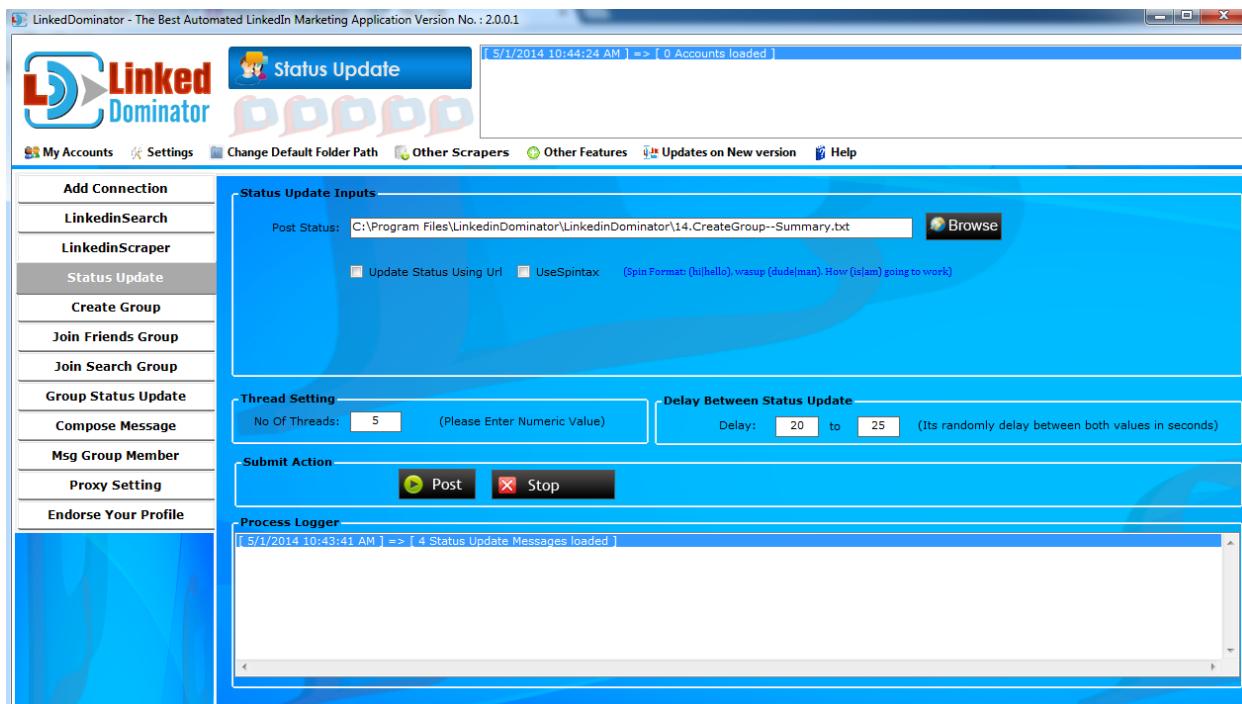
After login its passes all parameters in another method,  
“[StartCampaignLinkedinScraperWithPagination](#)”

```
StartCampaignLinkedinScraperWithPagination(ref HttpHelper, Account, FirstName,
LastName, Location, Country, LocationArea, PostalCode, Company, Keyword, Title,
IndustryType, Relationship, language, Groups, FileName, TitleValue, CompanyValue, within,
YearsOfExperience, Function, SeniorLevel, IntrestedIn, CompanySize, Fortune1000,
RecentlyJoined);
```

After that it's working same as “[LinkedinScrapper](#)” module.

### 1.8 Status Update

If you want to share your thoughts with the LinkedIn members then there is nothing better than updating your status in your LinkedIn profile.  
Status update help you with that.



- a) This is a feature that will post a update on your account.
- b) You need to write down the status one after another in a text file.
- c) The software will take the first line and will post the line in your first account uploaded.
- d) It works on one on one basis.
- e) But if you want to upload a link or a YouTube video in your account then you can select the check box "UpdateUsingURL" and the job is done.
- f) After the process is complete you will get the result file in the output folder in your desktop.



## How it work with Code,

Settings,

1. Thread Setting

Thread setting means its work with no of thread provided by user.

```
ThreadPool.SetMaxThreads(numberofThreds, numberofThreds);
ThreadPool.QueueUserWorkItem(new WaitCallback(PostStatus), new object[]
```

```
{ item });
Thread.Sleep(1000);
```

## 2. Delay Setting

Its randomly delay between threads after posting get/post request.

Basically its two values first delay and second delay.

In “**PostStatus**” method all setting are passes with object parameter,  
Login Process done by “**PostStatus**” method after login its call another method,

```
"PostStatusMsg",
obj_StatusUpdate.PostStatusMsg(ref HttpHelper, Statuswithurl, minDelay, maxDelay,
statusSpin, isSpinTrue);
```

finally message/status posting in method “**PostStatusMsg**” its available in class  
“**StatusUpdate**”

finally it is send status/message posting for user timeline,

### 1. First getrequest

```
string pageSource = HttpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/home?trk=hb_tab_home_top"));
```

after this request we will get csrfToken, sourceAlias

### 2. Final Post Request

```
string PostStatusData = string.Empty;
if (EntityId == string.Empty)
{
    PostStatusData = "ajax=true&contentImageCount=0&contentImageIndex=-1&contentImage=&contentEntityID=&contentUrl=&postText=" +
        Uri.EscapeDataString(Post) +
        "&contentTitle=&contentSummary=&contentImageIncluded=true&%23=&postVisibility2=EVERYONE&submitPost=&tetherAccountID=&tweetThisOn=false&postToMFeedDefaultPublic=true&csrfToken=" +
        csrfToken + "&sourceAlias=" + sourceAlias + "";
}
else
{
    PostStatusData = "ajax=true&contentImageCount=" + ImgCount +
        "&contentImageIndex=0&contentImage=" + LogoUrl + "&contentEntityID=" +
        EntityId + "&contentUrl=" + ReqUrl + "&postText=" + Uri.EscapeDataString(Post) +
        "&contentTitle=" + contentTitle + "&contentSummary=" + contentSummary +
        "&contentImageIncluded=true&%23=&postVisibility2=EVERYONE&submitPost=&tetherAccountID=&tweetThisOn=false&postToMFeedDefaultPublic=true&csrfToken=" +
        Uri.EscapeDataString(csrfToken) + "&sourceAlias=" + sourceAlias + "";
}

PostStatusData = PostStatusData.Replace(" ", "");
```

```
string ResponseStatusMsg = HttpHelper.postFormData(new
Uri("http://www.linkedin.com/nhome/submit-post"), PostStatusData);
```

The screenshot shows a LinkedIn profile page for 'swarupa rani'. A red arrow points to her post: "Hello Good Morning all of You." Below the post, there are options to Like, Comment, Share, and View 1s ago.

## 1.9 Create Group

Want to start a group then there is nothing better than create group feature to create groups of your choice in minutes. It will help you in creating many groups with different filters at a time.

The screenshot shows the LinkedDominator software interface. The main window title is "LinkedDominator - The Best Automated LinkedIn Marketing Application Version No.: 2.0.0.1". The "Create Group" tab is selected in the left sidebar. The main panel displays "5/1/2014 10:44:24 AM => [ 0 Accounts loaded ]". The "Create Group Inputs" section contains fields for Logo, GroupName, Summary, Description, Website, Access (Auto-Join or Request to Join), Creating Group (Create Open Group or Create Member Group), Language, GroupType, and Groups Per Account. Below this is a "Delay Between Create Group" section with a delay of 20 to 25 seconds. The "Submit Action" button is visible. The "Add Your Friends In Your Groups" section includes a "GroupURLs" field with a browse button and a "Send Invitation" button. The "Process Logger" section shows log entries: "[ 5/1/2014 10:43:41 AM ] => [ 2 Group Names loaded ]", "[ 5/1/2014 10:43:41 AM ] => [ 4 Group Summary loaded ]", and "[ 5/1/2014 10:43:41 AM ] => [ 3 Group Description loaded ]".

a)For creating group, it is very easy to do with LinkedIn Dominator.

- b) You just need to upload the necessary data written in a text file and feed them to the software. And after that Press "create group" button.
- c) It will create you the certain group with the features you wanted to be in your group.
- d) In a similar manner how all the modules get their output file, here also you will get the file created inside the folder "LinkedIn Dominator".
- e) It contains a special feature which is known as " Add you friends In Your Group"
- f) What happens here is you need to put the url of the group.
- g) Please make sure that where ever in the software you are uploading any group urls, it should be in this following format (<http://www.linkedin.com/groups?gid=2319076>)
- h) Every url you pick should contain till the gid (group id) number.
- i) Now this feature will send all your first connection invites to join the group you created.

**How it work with code,**

IF you add inputs like, Logo, Groupname, Summary, Description, Website, Access Type(AutoJoin, Request to join), Creating Group (OpenGroup, MemberGroup), Language, Group Type, Groups per account and delay setting, After that press "CreateGroup" button,  
First verify validation by user inputs before I discussed,  
After that execute method "[LinkdinCreateGroup\(\)](#)"  
Its passes another method with multithreaded environment

```
ThreadPool.SetMaxThreads(5, numberofThreds);
PostCreateGroup(new object[] { item });
Thread.Sleep(1000);
```

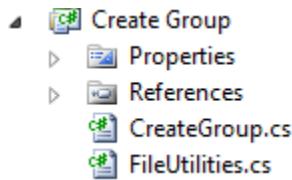
Login process done by method "[PostCreateGroup](#)"

```
if (!Login.IsLoggedIn)
{
    Login.LoginHttpHelper(ref HttpHelper);
}
```

After login it is passes another method with login ref look like below

```
obj_CreateGroup.StartCreateGroup(ref HttpHelper, minDelay, maxDelay);
```

CreateGroup Class contains all methods related to Create group



In process of method "StartCreateGroup"

1. First Get Request

```
PageSource1 = HttpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/createGroup?displayCreate=&trk=anet_creategrp"));
```

2. Post request for create group image

```
string result =
HttpHelper.HttpUploadFileBackground("http://www.linkedin.com/mupld/upload",
PostIamge, "file", "image/" + image, nvc, true, ref status);
```

3. Second Get Request for return for Group image thumbnail

```
string GetRequest = HttpHelper.getHtmlfromUrl1(new
Uri("http://media03.linkedin.com/media/" + TempID));
```

4. Second PostRequest for Create Member Group

```
PostCreateGroup = "csrfToken=" + Uri.EscapeDataString(csrfToken) +
"&acceptLogoTerms=acceptLogoTerms&groupName=" + PostGrpName +
"&groupCategory=" + SearchCriteria.GroupType +
"&otherGroupCategory=&shortDesc=" + PostGrpSummry + "&longDesc=" +
PostGrpDesc + "&homeSite=" + PostGrpWebsite + "&groupEmail=" +
Uri.EscapeDataString(accountUser) + "&groupInDirectory-open=groupInDirectory-
open&logoInProfiles-open=logoInProfiles-open&membersSendInvites-
open=membersSendInvites-open&access=request&groupInDirectory-
request=groupInDirectory-request&logoInProfiles-request=logoInProfiles-
request&emailDomains=&language=" + SearchCriteria.GroupLang +
"&countryCode=&postalCode=&acceptContract=acceptContract&create=Create+a+M
embers-Only+Group&gid=&largeLogoTempID=" + TempID +
"&discVisibility=false&tetherAccountId=&facebookTetherID=&uncroppedHeroImageI
D=&croppedHeroImageID=&heroImageCropParams=";
ResponseStatusMsg = HttpHelper.postFormDataRef(new
Uri("http://www.linkedin.com/createGroup"), PostCreateGroup,
"http://www.linkedin.com/createGroup", "", "");
```

Csv writing after creating Member Only group

```
string CSV_Content = accountUser + "," + PostGrpName + "," +
urlForNewGroupCreated;
CSVUtilities.ExportDataCSVFile(CSVHeader, CSV_Content, Globals.path_CreateGroups);
urlForNewGroupCreated = string.Empty;
```

5. Third Postrequest for open Group

```
PostCreateGroup = "csrfToken=" + Uri.EscapeDataString(csrfToken) +
"&acceptLogoTerms=acceptLogoTerms&groupName=" + PostGrpName +
"&groupCategory=" + SearchCriteria.GroupType +
"&otherGroupCategory=&shortDesc=" + PostGrpSummry + "&longDesc=" +
PostGrpDesc + "&homeSite=" + PostGrpWebsite + "&groupEmail=" +
Uri.EscapeDataString(accountUser) + "&access=open&groupInDirectory-
open=groupInDirectory-open&logoInProfiles-open=logoInProfiles-
open&membersSendInvites-open=membersSendInvites-open&groupInDirectory-
request=groupInDirectory-request&logoInProfiles-request=logoInProfiles-
request&emailDomains=&language=" + SearchCriteria.GroupLang +
"&countryCode=&postalCode=&acceptContract=acceptContract&create=Create+an+
Open+Group&gid=&largeLogoTempID=" + TempID +
"&discVisibility=true&tetherAccountID=&facebookTetherID=&uncroppedHeroImageI
D=&croppedHeroImageID=&heroImageCropParams=";
ResponseStatusMsg = HttpHelper.postFormDataRef(new
Uri("http://www.linkedin.com/createGroup"), PostCreateGroup,
"http://www.linkedin.com/createGroup?displayCreate=&displayCreate=&trk=hb_side
_cgrp", "", ""));
```

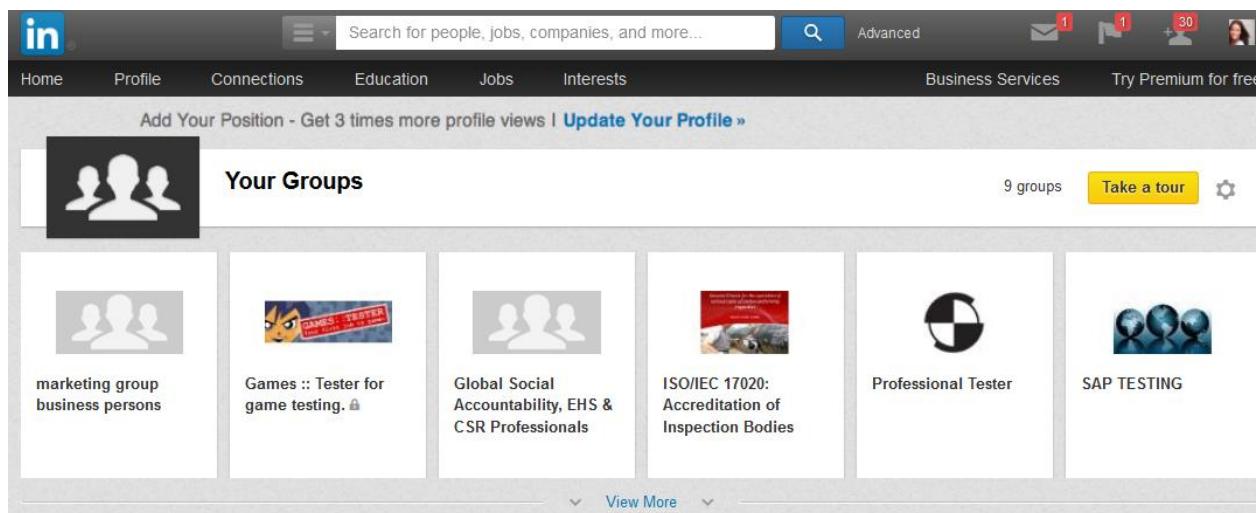
Csv writing after creating Open group

```
string CSV_Content = accountUser + "," + PostGrpName;
CSVUtilities.ExportDataCSVFile(CSVHeader, CSV_Content, Globals.path_CreateGroups);
```

All created groups save in table “tb\_LinkedinJoinGroup”

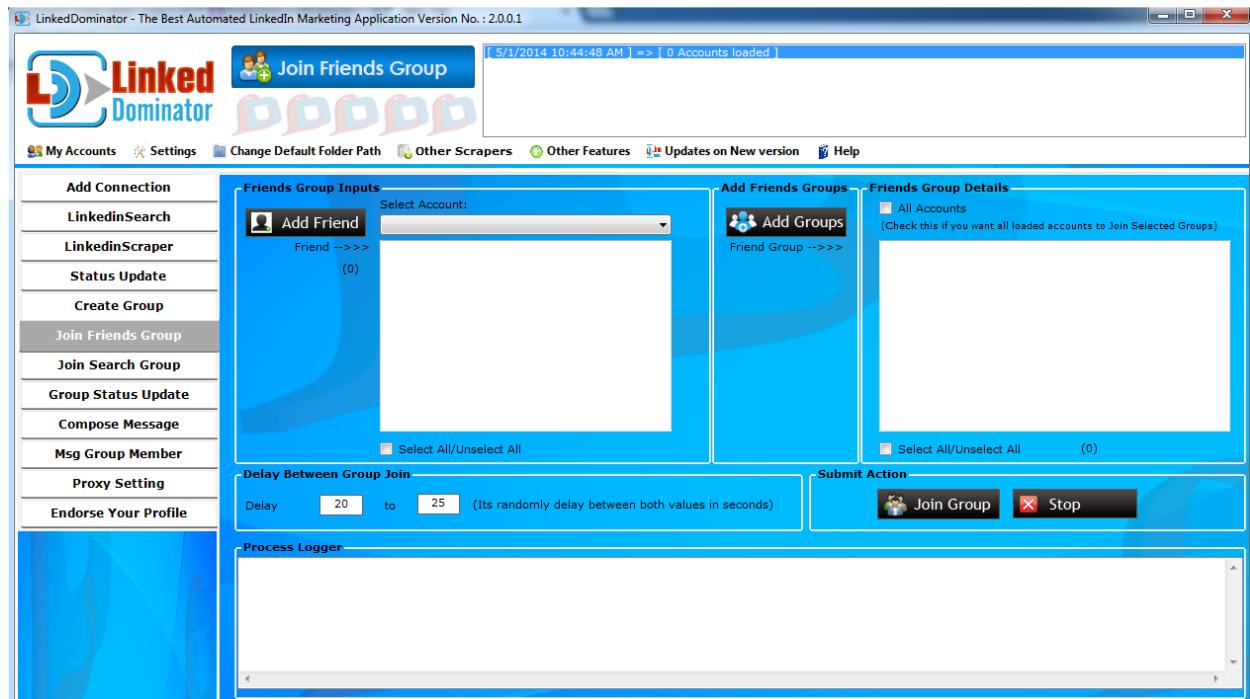
- ▷ tb\_CamapignSearcWithInvite
- ▷ tb\_CampaignJoinGroup
- ▷ tb\_CampaignScrap
- ▷ tb\_CampaignScrapURL
- ▷ tb\_LinkedInAccount
- ▷ tb\_LinkedinJoinGroup
- ▷ tb\_LinkedinScrapResult
- ▷ tb\_LinkedinSearchUrlResult
- ▷ tb\_ManageAddConnection
- ▷ tb\_ManageMsgGroupMem
- ▷ tb\_OnlyVisitProfile
- ▷ tb\_Proxies
- ▷ tb\_Scheduler\_Module
- ▷ tb\_SearchWithInvite

After creating group looks like in site pls see the below,



## 1.10 Join Friends Group

Want to join the groups which your friend has joined then you can have this feature and you can find out the full group list of your friend and then you can join those groups according to your choice



a)It's a very simple and powerful module.

b)What if you are following your friends in LinkedIn and you want to join your friend's all/specific groups.

c)This module "Join Friends Group" will help you with this.

- d)First click on “add friend” and wait for the process to get complete in the logger.
- e)Pressing that button will add all the friends of the accounts you uploaded in the software.
- f)Once the process is complete, select the email id from the drop down and then you will get the list of first connection in the box given.

### **How it work with code,**

Click Add friend, means all friends as your1st connection are added to friend list,  
After click “Add Friend” button,

It will passes “`LinkedInGroupMemberSearch()`” method,

Internally passes another method by “`LinkedInGroupMemberSearch()`” method,

```
if (LinkedInManager.linkedInDictionary.Count > 0)
{
    ThreadPool.SetMaxThreads(numberofThreads, 5);
    foreach (KeyValuePair<string, LinkedInMaster> item in
LinkedInManager.linkedInDictionary)
    {
        ThreadPool.SetMaxThreads(numberofThreads, 5);
        ThreadPool.QueueUserWorkItem(new
WaitCallback(StartDMMultiThreadedGroupMemmberAdd), new object[] { item });
        Thread.Sleep(2000);
    }
}
```

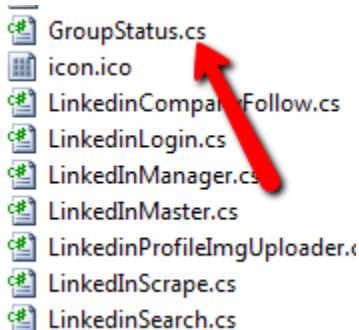
Method “`StartDMMultiThreadedGroupMemmberAdd`” working with multithreaded environment,

In this method login process is done,  
`Login.LoginHttpHelper(ref HttpHelper);`

After login process its passes another method which is return with `Dictionary<string, string>` mode,

```
Dictionary<string, string> Result = dataScrape.PostAddMembers(ref HttpHelper,
Login.accountUser);
LinkdInContacts.Add(Login.accountUser, Result);
```

“`PostAddMembers`” method are very important role to scraped all members contains by login user, which is contain in Class “`GroupStatus`”



1. First Get Request in "PostAddMembers" method

```
string pageSource = HttpHelper.getHtmlfromUrl(new
Uri("https://www.linkedin.com/contacts/api/"));
```

after scrapping all friends list added in with account wise,

```
LinkdInContacts.Add(Login.accountUser, Result);
```

After select combo box selectedindexChnaged event  
It's added all friend list in group list box

```
chkMembers.Items.Add(item1.Value.Replace(",", string.Empty));
```

After display all friend List it's ready to add Groups with selected members,

After that click "Add Groups" button

Its work with single threaded in method "LinkedInMembersGroupSearch()"

```
new Thread(() =>
{
    LinkedInMembersGroupSearch();
}).Start();
```

Method "StartDMMultiThreadedMemmbersGroupAdd" work with multithreaded environment,

```
foreach (KeyValuePair<string, LinkedInMaster> item in
LinkedInManager.linkedInDictionary)
{
    if (SelectedEmail.Contains(item.Key))
    {
        ThreadPool.SetMaxThreads(numberofThreads, 5);
        ThreadPool.QueueUserWorkItem(new
WaitCallback(StartDMMultiThreadedMemmbersGroupAdd), new object[] { item });
        Thread.Sleep(1000);
    }
}
```

}

Login process running with method “[StartDMMultiThreadedMemmbersGroupAdd](#)”

```
if (!Login.IsLoggedIn)
{
    Login.LoginHttpHelper(ref HttpHelper);
}
```

After completed the login process it's scraped all groups which is selected in friend list with login reference,

```
GroupStatus dataScrape = new GroupStatus();
Dictionary<string, string> Groups = dataScrape.PostAddGroupNames(ref HttpHelper,
MemId);
```

After properly scrapping added groups in List string var,

```
GrpAdd.Add(Login.accountUser, Groups);
```

Finally all groups are added GroupList Control

```
foreach (KeyValuePair<string, string> item1 in Grps.Value)
{
    try
    {
        chkExistGroup.Invoke(new MethodInvoker(delegate
        {
            chkExistGroup.Items.Add(item1.Key);
            UserID = cmbUser.SelectedItem.ToString();
        }));
    }
    catch (Exception ex)
    {
    }
}
```

After getting groups we are execute final process in submit action option

Click “Join Group” Button,

Its start with single threaded with method “[LinkedInAddGroups\(\)](#)”

```
new Thread(() =>
{
```

```
LinkedInAddGroups();  
}).Start();
```

After thread and delay setting its call passes other method with multithreaded environment,

```
ThreadPool.SetMaxThreads(numberOfThreads, 5);  
ThreadPool.QueueUserWorkItem(new WaitCallback(PostAddGroups), new object[] { item  
});  
Thread.Sleep(1000);
```

Login process is done in this method

```
GroupStatus obj_GroupStatus = new GroupStatus(Login.accountUser, Login.accountPass,  
Login.proxyAddress, Login.proxyPort, Login.proxyUserName, Login.proxyPassword);
```

```
if (!Login.IsLoggedIn)  
{  
    Login.LoginHttpHelper(ref HttpHelper);  
}
```

After successfully login its passes other method,

```
string MessagePosted = obj_GroupStatus.PostGroupAddFinal(Login.accountUser,  
Login.accountPass, minDelay, maxDelay);
```

After all final process done by method “**PostGroupAddFinal**”

1. First GetRequest

```
string pageSource = HttpHelper.getHtmlFromUrl1(new  
Uri("https://www.linkedin.com/home?trk=hb_tab_home_top"));
```

After this process we get csrfToken, sourceAlias, regCsrfParam

After getting all details for post request parameters,

2. First PostRequest

```
postUrl = "https://www.linkedin.com/uas/login-submit";  
postdata = "isJsEnabled=true&source_app=&tryCount=&session_key=" +  
Uri.EscapeDataString(Screen_name) + "&session_password=" + pass +  
&signIn=Sign%20In&session_redirect=&loginCsrfParam=" + regCsrfParam +  
&csrfToken=" + csrfToken + "&sourceAlias=" + sourceAlias;  
ResLogin = HttpHelper.postFormData(new Uri(postUrl), postdata);
```

Finally get request for check group successfully joined or not,

3. Second GetRequest

```
PostGroupstatus = post + itemGrps.Split('\"') [0].Replace("amp;",  
string.Empty).Replace("&goback=", "&trk=group-join-button").Replace(" ", "");  
ResponseStatusMsg = HttpHelper.getHtmlFromUrl1(new Uri(PostGroupstatus));
```

Delay setting for every other request with more than one account

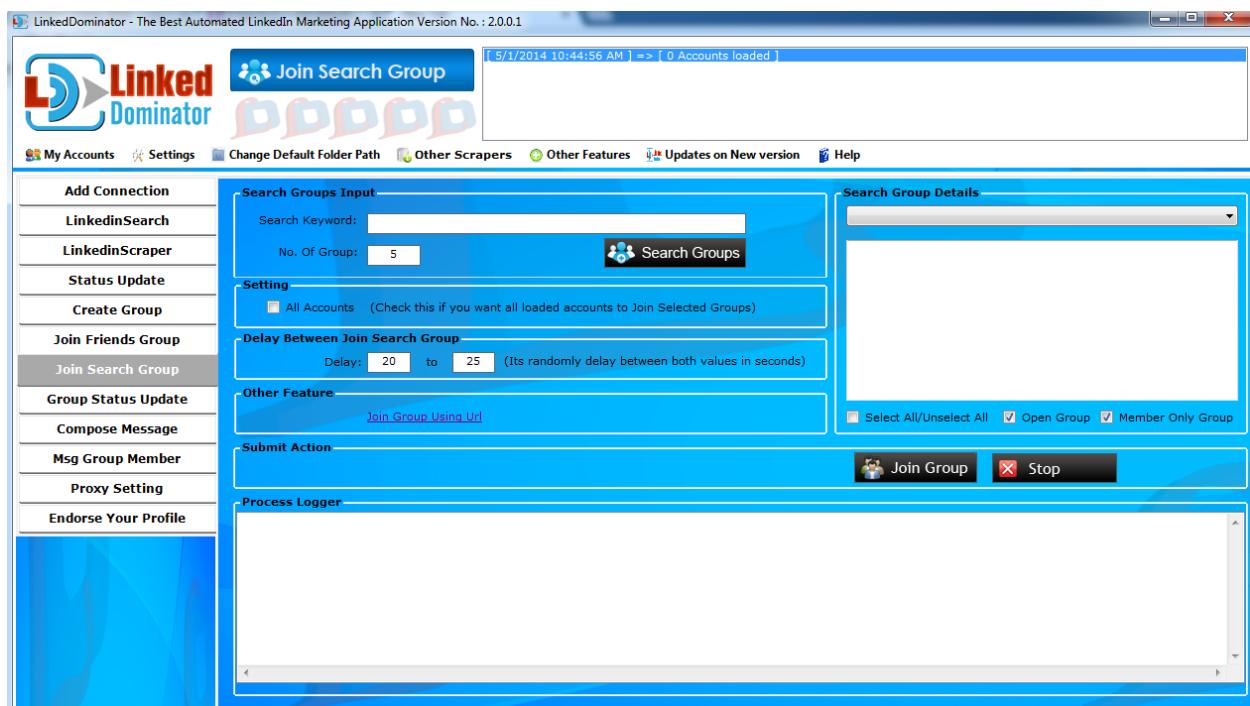
```
int delay = RandomNumberGenerator.GenerateRandom(mindelay, maxdelay);
Log("[ " + DateTime.Now + " ] => [ Delay for : " + delay + " Seconds ]");
Thread.Sleep(delay * 1000);
```

**Groups**

 <b>Mensa</b> 10,228 members <a href="#">+ Join</a>	 <b>HP Alumni</b> - Hewlett... 20,167 members <a href="#">+ Join</a>	 <b>On Startups</b> - The Co... 441,855 members <a href="#">+ Join</a>	 <b>UT Austin Alumni Gr...</b> 9,359 members <a href="#">+ Join</a>
 <b>Former Employees o...</b> 2,627 members <a href="#">+ Join</a>	 <b>Disney</b> 20,287 members <a href="#">+ Join</a>	 <b>McCombs MBA Alum...</b> 7,387 members <a href="#">+ Join</a>	 <a href="#">See 40 more &gt;</a>

### 1.11 Join Search Group

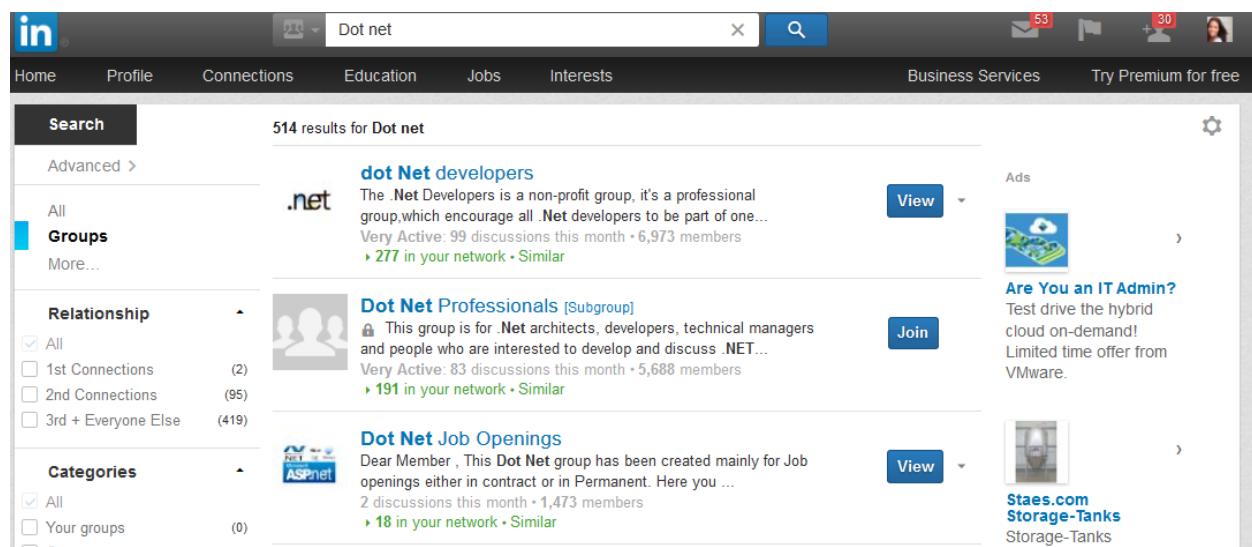
Don't have time to search group of your choice manually then you can use this feature where you need to type in the keyword of your choice according to which software will return you the list of groups. Then you can join any of them according to your choice.



- a) If you want to join some groups related to some particular niche then it would be very easy for your accounts to search and join them using this module.
- b) Type in the word in the "Search Keyword" whose groups you want to join and specify the number of groups you want the software to find for you.
- c) Press "Search Groups" button and then it will find you those many number of groups related to that keyword in the box given in right side.
- d) After you select the group which you want to join you can press the "Join Group" button and then the accounts uploaded will go and join the groups you selected above.
- e) As like the other modules, this module will also give you the output file with all the accounts loaded and which group you joined will be there.
- f) The output folder will contain two files, one will indicate the file containing the list of groups joined and the other file will contain the list of groups that were not able to joined.

How it works with code,

Actually we have search in LinkedIn site looks like below,



The screenshot shows the LinkedIn search interface with the query 'Dot net'. The results page displays three main group entries:

- dot Net developers**: A non-profit group for .Net developers, with 514 results.
- Dot Net Professionals [Subgroup]**: A subgroup for .Net architects, developers, and managers, with 191 results.
- Dot Net Job Openings**: A group for job openings, with 18 results.

On the left sidebar, filters for Relationship (All, 1st Connections, 2nd Connections, 3rd + Everyone Else) and Categories (All, Your groups) are visible. On the right, there are ads for 'Are You an IT Admin?' and 'Staes.com Storage-Tanks'.

In technically there are four modules we have covered with coding,

1. Search Group Input
2. Delay setting
3. Search Group Details
4. Submit action

#### 1. Search Group Input

In this module we are added keyword according to group search after that we will click "Search Groups" button after click its start single threaded with method, "LinkedInOpenGroupSearch()"

```
new Thread(() =>
{
    LinkedInOpenGroupSearch();
}).Start();
```

After that its passes another method with multithreaded environment in Method "StartDMMultiThreadedSearchOpenGroups"

```
ThreadPool.SetMaxThreads(numberofThreads, 5);
ThreadPool.QueueUserWorkItem(new
WaitCallback(StartDMMultiThreadedSearchOpenGroups), new object[] { item });
Thread.Sleep(1000);
```

Login process running in this method

```
GlobusHttpHelper HttpHelper = new GlobusHttpHelper();
```

```
if (!Login.IsLoggedIn)
{
    Login.LoginHttpHelper(ref HttpHelper);
}
```

After completed Login process its scrapped group with keyword search wise,

```
Result = obj_JoinSearchGroup.PostAddOpenGroups(ref HttpHelper,
txtSearchKeyword.Text.ToString().Trim(), item.Value._Username);
```

```
int count = 5;
if (!string.IsNullOrEmpty(textBoxGroupCount.Text) &&
NumberHelper.ValidateNumber(textBoxGroupCount.Text))
{
    count = Convert.ToInt32(textBoxGroupCount.Text);
}
```

```
LinkdInContacts.Add(Login.accountUser, Result);
```

After collecting var "LinkdInContacts" its display groups combo

```
if (cmbSearchGroup.InvokeRequired)
{
    new Thread(() =>
    {
        cmbSearchGroup.Invoke(new MethodInvoker(delegate
        {
            cmbSearchGroup.Items.Add(Login.accountUser);
        }));
    }).Start();
}
```

After display all groups in Combo box we are click combo box with event  
"cmbSearchGroup\_SelectedIndexChanged"

In this section display groups with user wise,

```
foreach (KeyValuePair<string, string> item1 in item.Value)
{
    AddLoggerSearchGroup("[ " + DateTime.Now + " ] => [ " + item1.Key + " ]");
    chkListSearchGroup.Items.Add(item1.Key);
}
```

After added groups account wise we will click "Join Group" Button for final process,

It is work with single threaded in method “[LinkedInAddSearchGroups\(\)](#)”

```
new Thread(() =>
{
    LinkedInAddSearchGroups();
}).Start();
```

After setting threads and delay setting its passes other method with multithreaded environment, “[PostAddSearchGroups](#)”

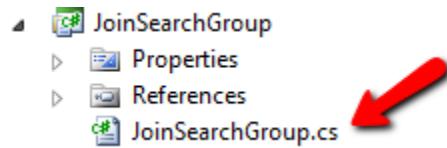
```
ThreadPool.SetMaxThreads(numberOfThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(PostAddSearchGroups), new object[] {
item });
Thread.Sleep(1000);
```

Method “[PostAddSearchGroups](#)” is responsible for login below code snippet,

```
JoinSearchGroup.JoinSearchGroup obj_JoinSearchGroup = new
JoinSearchGroup.JoinSearchGroup(Login.accountUser, Login.accountPass,
Login.proxyAddress, Login.proxyPort, Login.proxyUserName, Login.proxyPassword);

if (!Login.IsLoggedIn)
{
    Login.LoginHttpHelper(ref HttpHelper);
}
```

After successfully login its passes other method “[PostSearchGroupAddFinal](#)” which is located in separate class “[JoinSearchGroup](#)”



```
string MessagePosted = obj_JoinSearchGroup.PostSearchGroupAddFinal(ref HttpHelper,
Login.accountUser, Login.accountPass, SelectedItem, minDelay, maxDelay);
```

1. 1<sup>st</sup> GetRequest

```
string pageSource = HttpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/home?trk=hb_tab_home_top"));
```

After get request we are getting, csrfToken, sourceAlias for post request

2. 1<sup>st</sup> PostRequest

```
postUrl = "https://www.linkedin.com/uas/login-submit";
```

```
postdata = "session_key=" + Screen_name + "&session_password=" + pass +
"&source_app=&trk=guest_home_login&session_redirect=&csrfToken=" + csrfToken
+ "&sourceAlias=" + sourceAlias;
ResLogin = HttpHelper.postFormData(new Uri(postUrl), postdata);
```

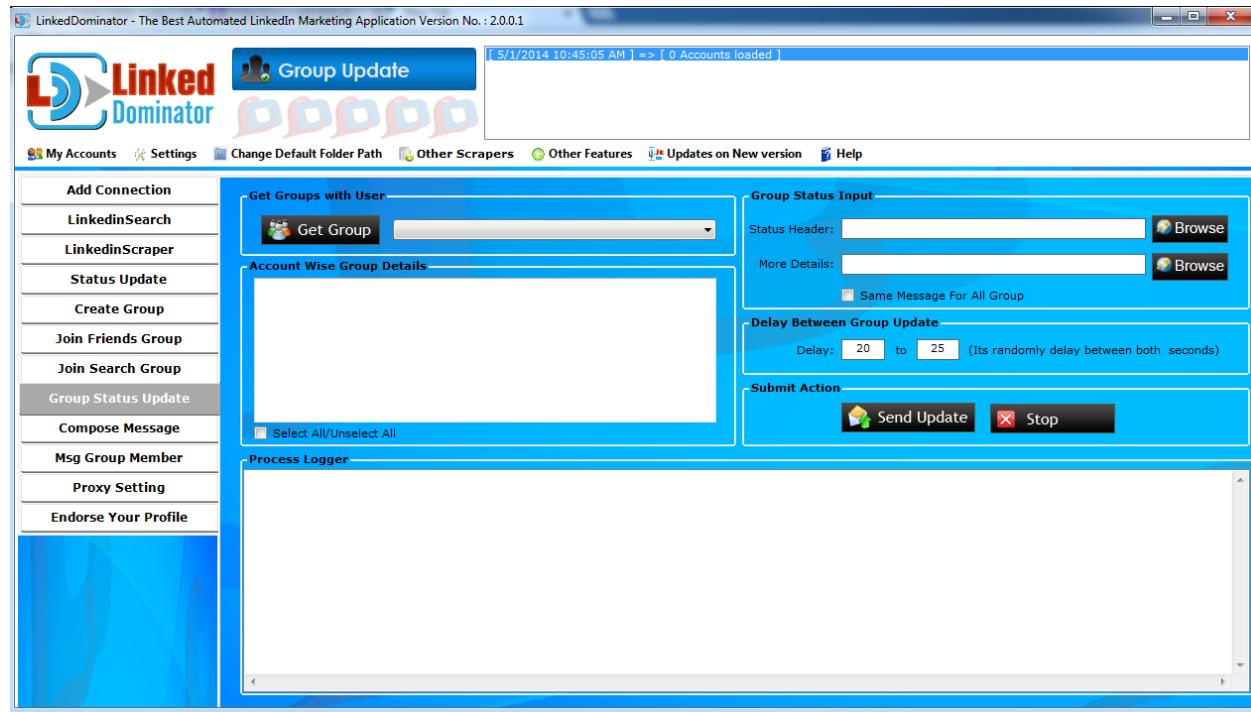
### 3. 2<sup>nd</sup> getrequest for check Group Successfully joined or not

```
string pageGetreq = HttpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/groupRegistration?gid=" + Itemgid.Split(':')[1] +
"&csrfToken=" + csrfToken + "&trk=group-join-button"));
```

## 1.12 Group Status Update

Want to share or promote something with the group then you can have them share with your group using this feature.

It update all your status at the respective group at a time. [www.linkeddominator](http://www.linkeddominator.com).



- a) This is the module which will help you in getting the status updated in the groups in which you are a member.
- b) It is very simple as like compared to other modules.
- c) You just need to write down "Status Header" and "More Details" in two separate text files and upload them in the respective text fields.
- d) Let's start with pressing "Get Group" button and wait for the process to get complete.
- e) After you get the list of groups, select the group to which you want to post a update.
- f) Upload the necessary txt files in 'Status Header' and 'More Details' section.
- g) Press 'Send Update' button and wait for the process to get complete.

h)Once the message is posted in the group, you will get a message saying the process is complete and there will be a file generated in desktop with the process details.

### How it work with code,

There 4 module,

1. Get group details with account wise.
2. Group Status input which input Status Header, Status Details with multiple manner means we have add more than one status header and details with browsing.
3. Delay and thread setting.
4. Submit action final process.

1. Get group details with account wise.

Start with click "Get Group" button start single threaded with method  
**"LinkdinGroupUpdate()"**

```
new Thread(() =>
{
    LinkdinGroupUpdate();
}).Start();
```

After setting its passes another method "**StartDMMultiThreadedGroupUser**"

```
ThreadPool.SetMaxThreads(numberofThreds, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(StartDMMultiThreadedGroupUser), new
object[] { item });
Thread.Sleep(1000);
```

Login process is running in method "**StartDMMultiThreadedGroupUser**"

```
GroupUpdate.GroupUpdate obj_GroupUpdate = new
GroupUpdate.GroupUpdate(Login.accountUser, Login.accountPass, Login.proxyAddress,
Login.proxyPort, Login.proxyUserName, Login.proxyPassword);

if (!Login.IsLoggedIn)
{
    Login.LoginHttpHelper(ref HttpHelper);
}
```

After login its scrapped groups according to Account and added to global variable

```
Data = obj_GroupUpdate.PostCreateGroupNames(ref HttpHelper, Login.accountUser);
GrpMess.Add(Login.accountUser, Data);
```

After collecting all groups' method is added group in combo with accounts wise

```
new Thread(() =>
```

```
{
    cmbGroupUser.Invoke(new MethodInvoker(delegate
    {
        cmbGroupUser.Items.Add(Login.accountUser);
    }));
}.Start();
```

Then we select group which have updated with status header and status details,

Added both values with browsing, set min and max delay,

After this we will click “Send Update” button which is available in final submit option,

Final process is start with single threaded,

```
new Thread(() =>
{
    LinkedInGroupMessage();
}).Start();
```

After that its passes another method “**PostGroupMsgUpdate**” with multithreaded manner,

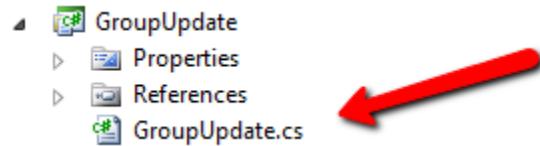
```
ThreadPool.SetMaxThreads(numberOfThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(PostGroupMsgUpdate), new object[] {
item });
Thread.Sleep(1000);
```

Login process is running in this method

```
GroupUpdate.GroupUpdate obj_GroupUpdate = new
GroupUpdate.GroupUpdate(Login.accountUser, Login.accountPass, Login.proxyAddress,
Login.proxyPort, Login.proxyUserName, Login.proxyPassword);

Login.LoginHttpHelper(ref HttpHelper);
```

After successfully Login this method passes another method  
**“PostGroupSameMessageForAllGroup”** which is available in separate class “**GroupUpdate**”



```
obj_GroupUpdate.PostGroupSameMessageForAllGroup(ref HttpHelper, SelectedItem, new
object[] { item, user }, minDelay, maxDelay);
```

In this method first Get Request

```
pageSource = HttpHelper.getHtmlfromUrl1(new  
Uri("http://www.linkedin.com/home?trk=hb_tab_home_top"));
```

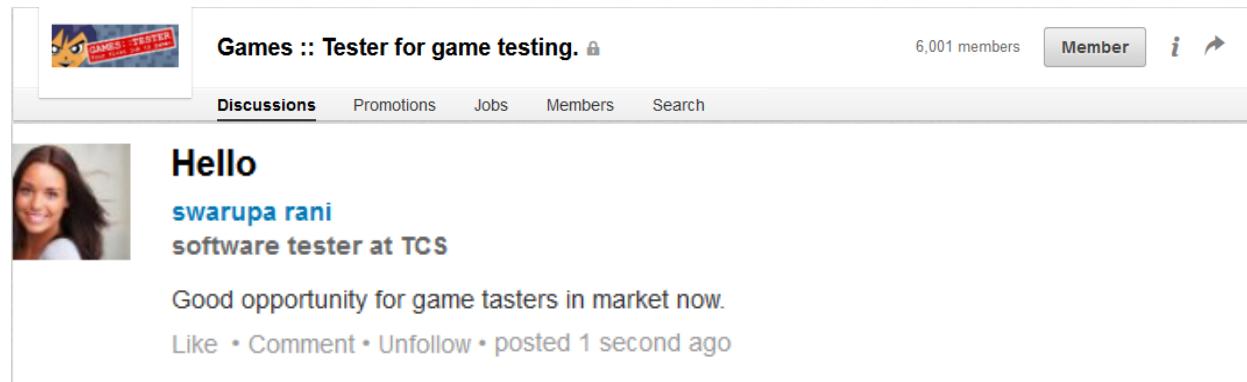
After first Get Request we will get csrfToken, sourceAlias

After getting parameter detail for final Post Request,

```
PostGroupstatus = "csrfToken=" + csrfToken + "&postTitle=" + PostGrpDiscussion +  
&postText=" + PostGrpMoreDetails + "&pollChoice1-ANetPostForm=&pollChoice2-  
ANetPostForm=&pollChoice3-ANetPostForm=&pollChoice4-ANetPostForm=&pollChoice5-  
ANetPostForm=&pollEndDate-ANetPostForm=0&contentImageCount=" + ImgCount +  
&contentImageIndex=-1&contentImage=" + LogoUrl + "&contentEntityID=" + EntityId +  
&contentUrl=" + ReqUrl + "&contentTitle=" + contentTitle + "&contentSummary=" +  
contentSummary + "&contentImageIncluded=true&%23=&gid=" + Gid +  
&postItem=&ajax=true&tetherAccountID=&facebookTetherID=";
```

```
ResponseStatusMsg = HttpHelper.postFormData(new  
Uri("http://www.linkedin.com/groups"), PostGroupstatus);
```

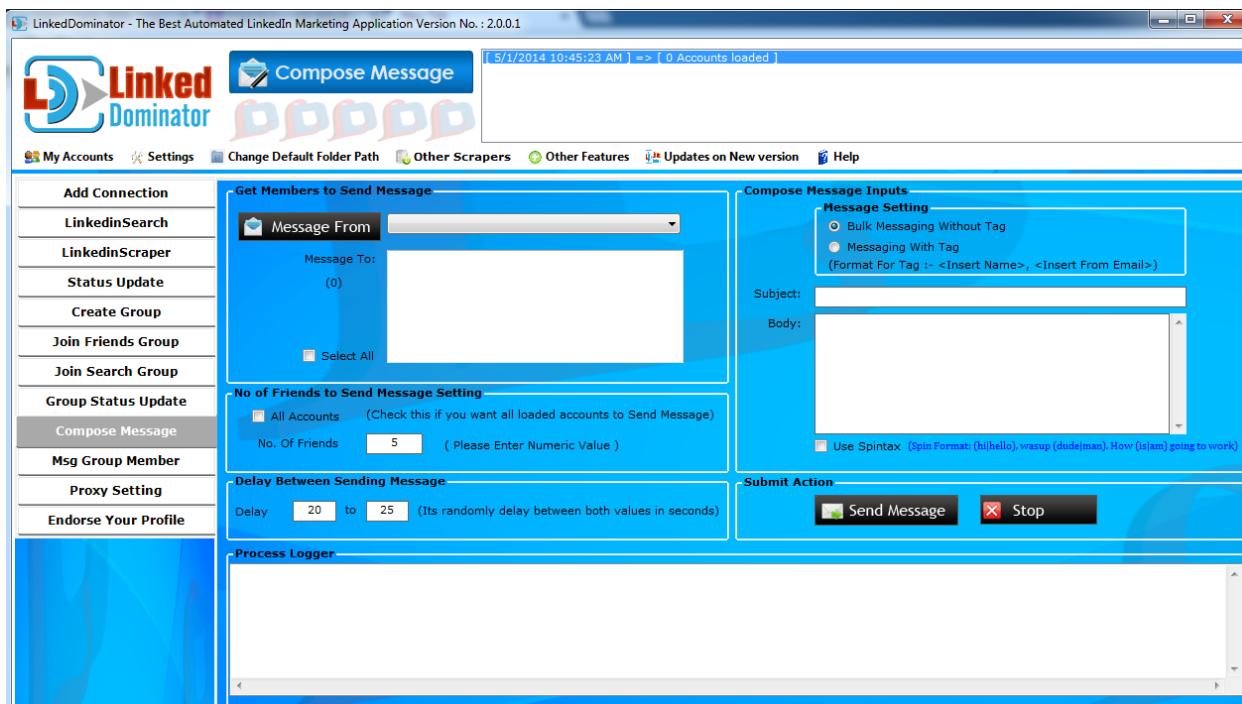
Result looks like below,



The screenshot shows a LinkedIn group page for 'Games :: Tester for game testing.' with 6,001 members. A post from user 'swarupa rani' (software tester at TCS) is displayed, reading: 'Hello' and 'Good opportunity for game tasters in market now.' The post was made 1 second ago and has 0 likes, 0 comments, and 0 unfollows.

### 1.13 Compose Message

Want to do bulk message to all of your connections and you don't have time for that then you can compose message feature



- We will begin with fetching the 1st connection list from a user profile.
- Press 'Message From' button at first and wait for the process to get complete.
- Once the process gets complete you will get the list of all the 1st connection list in the window placed below.
- Now from there you can select the connection or all connection to which you want to send message.
- Once you have done that you can now go to the 'Compose Message Input' section and there you can write in the message Subject and Body.

**Note:**

- We have option of spintax text which will help you in sending unique message to each and every connection you want to receive the message sent by you.
- You can also use the message setting where in you select the option to send message with tag or without tag.

**How it work with code,**

There are three modules in technically divided this feature,

1. Get Members to send message
  2. Compose message inputs
  3. Submit action
1. Get Members to send message

This is start with method “[LinkedInAddFromID\(\)](#)” with single threaded

```
new Thread(() =>
{
    LinkdinAddFromID();
}).Start();
```

During this method passes another method “**StartDMMultiThreadedComposeMessage**” with multithreaded manner,

```
ThreadPool.SetMaxThreads(5, 5);
ThreadPool.QueueUserWorkItem(new
WaitCallback(StartDMMultiThreadedComposeMessage), new object[] { item });
Thread.Sleep(1000);
```

Login process done during run in method “**StartDMMultiThreadedComposeMessage**”

```
Login.LoginHttpHelper(ref HttpHelper);
```

After Login its passes another method for collect Friend List

```
Normal condition,
Dictionary<string, string> Result = MemberScrape.PostAddMembers(ref HttpHelper,
Login.accountUser);
MessageContacts.Add(Login.accountUser, Result);
```

```
With excel input conditions,
Dictionary<string, string> Result = MemberScrape.PostaddMembersWithExcelInput(ref
HttpHelper, Login.accountUser);
MessageContacts.Add(Login.accountUser, Result);
```

After scrapping friend List its added Combo box in event  
“**cmbMsgFrom\_SelectedIndexChanged**”

```
foreach (KeyValuePair<string, string> item1 in item.Value)
{
    string group = item1.Key;
    string[] group1 = group.Split(':');

    if (GetUserID == item.Key.ToString())
    {
        chkMessageTo.Items.Add(item1.Value.Replace(", ", ""));
        GmUserID.Add(item1.Value.Replace(", ", ""));
    }
}
```

## 2. Compose Mgs Input

There are two option to send mgs 1. Bulk Mgs with Tag 2. Mgs with tag.

Put subject and Body text.

Spin text is optional.

After that setting we are click final process for button "**Send Message**"

This method is starting with basic settings after that its call method  
"**LinkedInComposeMessage**"

```
new Thread(() =>
{
    LinkedInComposeMessage();
}).Start();
```

After process during call another method "**PostMessageBulk**" with multithreaded manner

```
ThreadPool.SetMaxThreads(numberOfThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(PostMessageBulk), new object[] { item });
Thread.Sleep(1000);
```

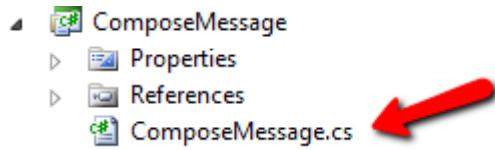
Login Process running during this method

```
if (!Login.IsLoggedIn)
{
    Login.LoginHttpHelper(ref HttpHelper);
}
```

After Login its get all details of User Mail web ID and Username with login reference,

```
string FromEmailId = MemberScrape.FromEmailCodeComposeMsg(ref HttpHelper, Userid);
string FromEmailName = MemberScrape.FromName(ref HttpHelper);
```

After that its call another important method there are two options both methods are working same manner only some setting difference which is available in separate class "**ComposeMessage**"



1. With Tag

```
obj_ComposeMessage.PostFinalMsg_1By1(ref HttpHelper, SelectedContacts,
subjectlistCompose, msgBodycomposePass, msgSubCompose.ToString(),
msgBodyCompose.ToString(), UserEmail, FromEmailId, FromEmailName, msg_spintaxt,
minDelay, maxDelay);
```

2. Without Tag

```
obj_ComposeMessage.PostFinalMsg(ref HttpHelper, SelectedContacts,
msgBodycomposePass, subjectlistCompose, msgSubCompose.ToString(),
msgBodyCompose.ToString(), UserEmail, FromEmailId, FromEmailName, msg_spintaxt,
minDelay, maxDelay);
```

1. First Get Request

```
string pageSource = HttpHelper.getHtmlFromUrl1(new
Uri("http://www.linkedin.com/home?trk=hb_tab_home_top"));
```

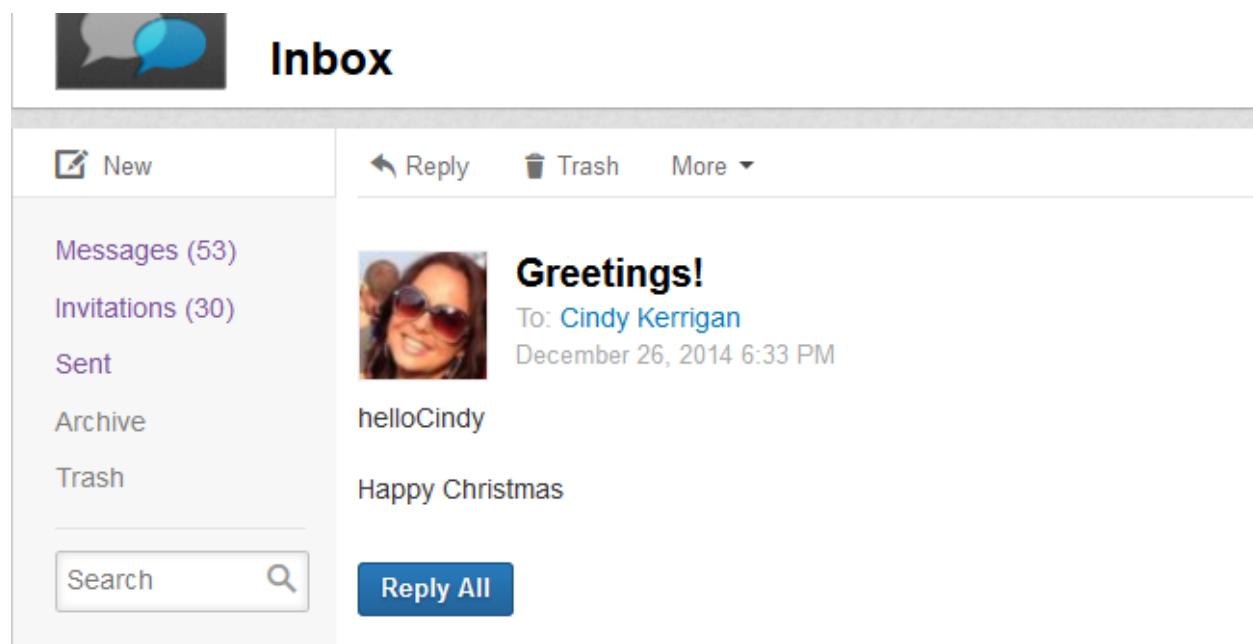
After getting all detail with first Get request we will post, Post Request

2. Final Post Request

```
tempBody = tempBody.Replace("<Insert Name>", string.Empty).Replace("<Insert
Group>", string.Empty).Replace("<Insert From Email>", string.Empty);
PostMessage = "csrfToken=" + csrfToken + "&subject=" +
Uri.EscapeDataString(tempSubject.ToString()) + "&body=" +
Uri.EscapeDataString(tempBody.ToString()) + "&submit=Send+Message&fromName=" +
FromEmailNam + "&fromEmail=" + FromEmailId + "&connectionIds=" + connId +
"&connectionNames=" + Uri.EscapeUriString(Nstring) +
"&allowEditRcpt=true&addMoreRcpt=false&openSocialAppBodySuffix=&st=&viewerDestin
ationUrl=&goback=.smg_*1_*1_*1_*1_*1_*1_*1_*1_*1";
```

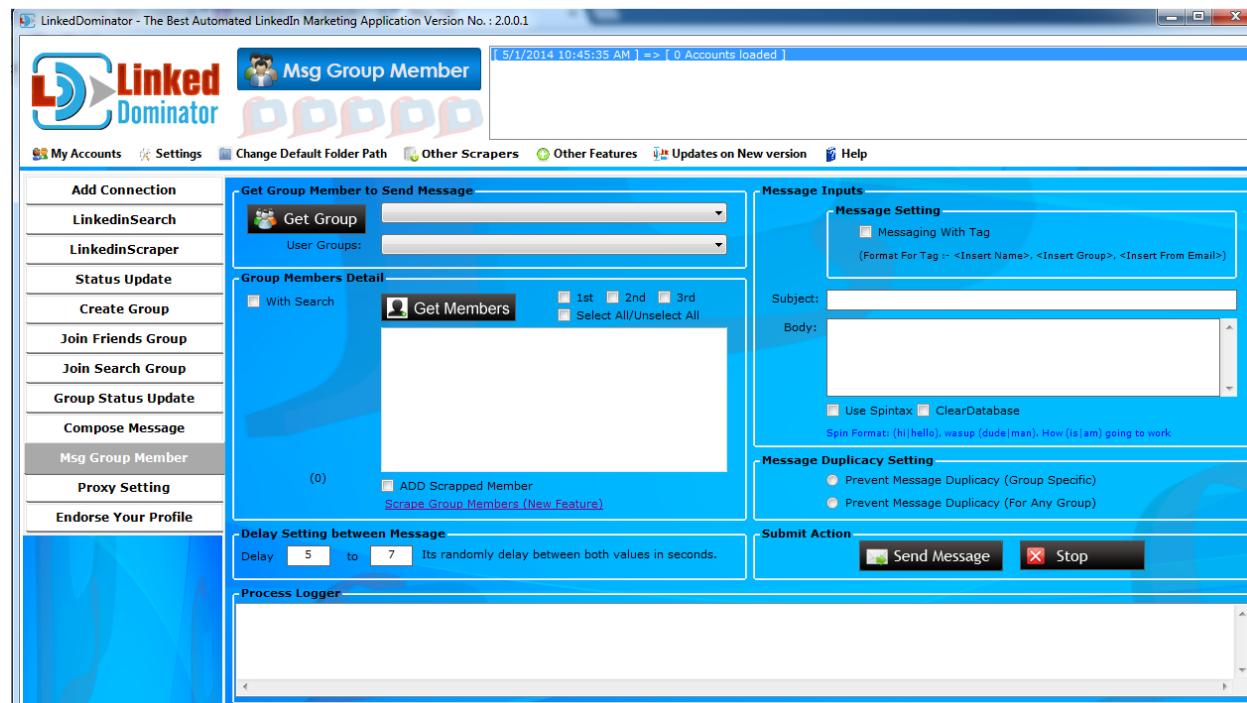
```
ResponseStatusMsg = HttpHelper.postFormData(new
Uri("http://www.linkedin.com/msgToConns"), PostMessage);
```

Finally result looks like in LinkedIn site mgs section,



### 1.14 Message Group Member

Want to message to people who are not in your contacts then you can use this feature which will help you to message group members at a time and that too in bulk. It also have a advantage that don't allow a member to receive duplicate messages.



- a)One of the powerful and effective tools in the LinkedIn market which allows you to send bulk/personal message to your group members even if they are not in your connection.

- b)The two things which we will be needing to start the process are the groups and their members. Software will get those groups who are in active status and you are a member of it.
- c) Lets start with by getting the list of groups of the account you have uploaded.
- d)Press 'Get Group' and wait for the software to fetch the list of group.
- e)In the logger you will get a message which will opt you to select the account.
- f)Select any particular account and then select the group from the list of group that will come at the dropdown.
- g)Once you have selected the account and its group, press 'Get Member' button and wait for the process to complete.
- h)Once the process gets complete you will get a message in logger that will opt you to select the members to whom you want to send message.
- i)Select the list of members to whom you want to send message.
- j)Now move to 'Message Input' section and write down the message subject and message body. After you have done that press 'Send Message' button.

Note: We have two option that avoid a member from getting a message twice, to avoid spamming.

- 1.Prevent Message Duplicacy (Group Specific)- this will avoid a member from getting the same message twice in a particular group.
- 2.Prevent Message Duplicacy (For Any Group) – this will avoid a member from getting a same message twice in any group.

 MessageSentGroupMember	4/26/2014 12:47 PM	Microsoft Excel C...	1 KB
--	--------------------	----------------------	------

- h)The output file will contain the list of accounts along with the message sent to it.

### How it work with code:

There are different modules in Mgs Group members,

1. Get Groups details with account wise.
2. Get members/friend list with group wise also customized based on excel inputs url and also search wise.
4. Compose message inputs
5. Submit action

1. Get Groups details with account wise.

This is start with method "[LinkdinGroupMemberUpdate\(\)](#)" with single threaded

LinkedDominator Document Ver 1.0

```
new Thread(() =>
{
    LinkdinGroupMemberUpdate();
}).Start();
```

After process during call another method “**StartDMMultiThreadedGroupMemberUser**” with multithreaded manner

```
ThreadPool.SetMaxThreads(numberofThreds, 5);
ThreadPool.QueueUserWorkItem(new
WaitCallback(StartDMMultiThreadedGroupMemberUser), new object[] { item });
Thread.Sleep(1000);
```

Login process is running in method “**StartDMMultiThreadedGroupMemberUser**”  
After login its get all group with accounts wise

```
Dictionary<string, string> Data = dataScrape.PostCreateGroupNames(ref HttpHelper,
Login.accountUser);
GrpMemMess.Add(Login.accountUser, Data);
```

After getting all group details its added combo box control with single threaded,

```
new Thread(() =>
{
    cmbAllUser.Invoke(new MethodInvoker(delegate
    {
        if (!cmbAllUser.Items.Contains(Login.accountUser))
        {
            cmbAllUser.Items.Add(Login.accountUser);
        }
    }));
}).Start();
```

After collecting group with user wise we have select user for display groups,

```
cmbMemberGroup.Items.Add(group1[1] + ':' + group1[0].ToString());
GroupStatus.GroupMemUrl.Add(item1.Key + ":" + item1.Value);
```

After select group we are ready to scrape all members/friend list with group wise,

2. Get Group member  
This is start with method “**LinkedAddSpecifiedGroupMem()**” with single threaded

```
new Thread(() =>
{
    LinkedAddSpecifiedGroupMem();
}).Start();
```

Process during method “[LinkedInComposeMessageGroupMem\(\)](#)” Its call other method “[StartCrawlSpecificGroupUser](#)”

```
StartCrawlSpecificGroupUser(new object[] { item });
```

Process during method “[StartCrawlSpecificGroupUser](#)” after login

```
Login.LoginHttpHelper(ref HttpHelper);
```

Get group member

```
Dictionary<string, string> GroupMem = dataScrape.AddSpecificGroupUser(ref HttpHelper,  
Login.accountUser, SpeGroupId1);
```

With excel input,

```
GroupMemData = dataScrape.AddSpecificGroupUserWithExcelInput(ref HttpHelper,  
Login.accountUser, SpeGroupId);
```

### 3. Message Inputs

There are one option to send mgs with tag.

Put subject and Body text.

Spin text is optional.

Mgs Duplicity setting,

1. Group Specific
2. For any group
3. Global

After that setting we are click final process for button “[Send Message](#)”

This method is starting with basic settings after that its call method “[LinkedInComposeMessageGroupMem\(\)](#)”

```
new Thread((() =>  
{  
    LinkedInComposeMessageGroupMem();  
}).Start());
```

After process during method “[LinkedInComposeMessageGroupMem](#)” its run another method “[PostMessageGroupMembers](#)”

```
PostMessageGroupMembers(new object[] { item });
```

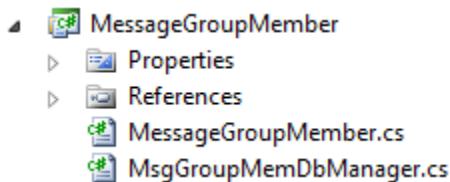
Process during method “[PostMessageGroupMembers](#)” Login process done in this method,

```
if (!Login.IsLoggedIn)  
{  
    Login.LoginHttpHelper(ref HttpHelper);  
}
```

After successfully login its run final method “**PostFinalMsgGroupMember\_1By1**”

```
obj_MessageGroupMember.PostFinalMsgGroupMember_1By1(ref HttpHelper,  
SelectedGroupMem, GrpMemSubjectlist, GrpMemMessagelist, msgSub, msgBody,  
selectedusername, FromemailId, FromEmailNam,  
_SelectedGroupName_MessageGroupMember, SpeGroupId, mesg_with_tag, msg_spintaxt,  
minDelay, maxDelay, preventMsgSameGroup, preventMsgWithoutGroup,  
preventMsgGlobal);
```

This method is contain separate class “**MessageGroupMember**”



Process during method “**PostFinalMsgGroupMember\_1By1**”

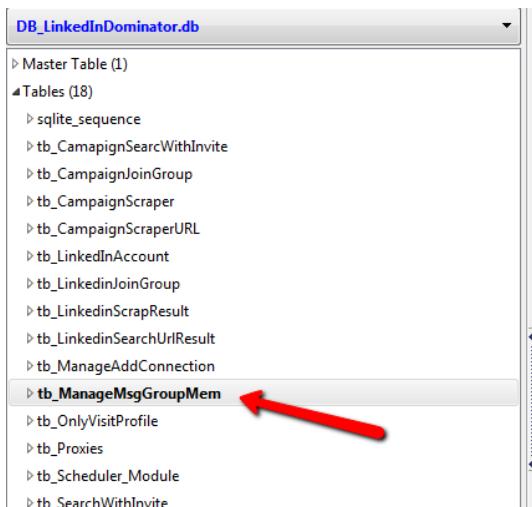
First get request,

```
string getComposeData = HttpHelper.getHtmlfromUrl(new  
Uri("https://www.linkedin.com/inbox/compose"));
```

After first get request get csrfToken, sourceAlias

Duplicity setting it's maintain separate table,

```
string Querystring = "Select  
MsgFrom,MsgToId,MsgTo,MsgGroupId,MsgGroupName,MsgSubject,MsgBody From  
tb_ManageMsgGroupMem Where MsgFrom ='" + UserName + "' and MsgGroupId = " +  
grpId + " and MsgToId = " + connId + "";  
ds = BaseLib.DataBaseHandler.SelectQuery(Querystring, "tb_ManageMsgGroupMem");
```



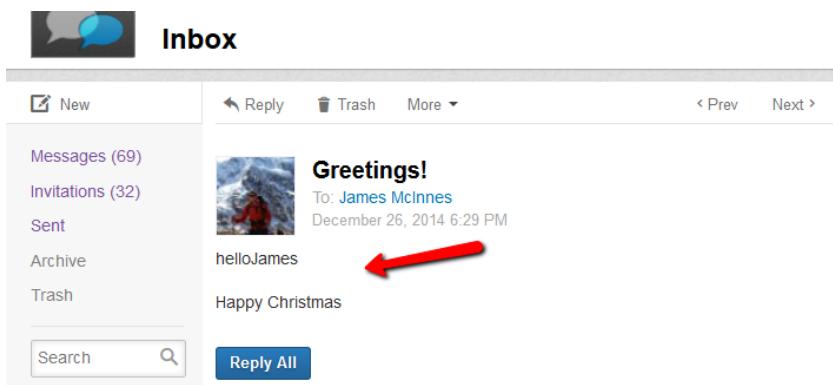
Id	MsgFrom	MsgTo	MsgTold	MsgGroupId	MsgGroup
1	Swarupa123@o...	sameerasingh(1...	381793325	381793325	Professional
2	Swarupa123@o...	RobWirsycz(2n...	126899	126899	Professional
3	Swarupa123@o...	sameerasingh(1...	381793325	381793325	Software te...
4	Swarupa123@o...	sameera singh(...	381793325	381793325	Software te...
5	Swarupa123@o...	sameera singh(...	381793325	381793325	Professional
6	Swarupa123@o...	Razvan M.(3rd)	368523100	368523100	Professional
7	Swarupa123@o...	Iuliana S.(3rd)	378843682	378843682	Professional

Final post Request,

```
PostMessage = "csrfToken=" + csrfToken + "&subject=" +
Uri.EscapeDataString(msg.ToString()) + "&body=" +
Uri.EscapeDataString(tempBody.ToString()) + "&submit=Send+Message&fromName=" +
Uri.EscapeDataString(FromEmailNam) + "&showRecipients=showRecipients&fromEmail=" +
FromemailId + "&connectionIds=" + connId +
"&connectionNames=&allowEditRcpt=true&addMoreRcpt=false&openSocialAppBodySuffix
=&st=&viewerDestinationUrl=&contentType=MEBC&groupID=" + grpId + "";
```

```
ResponseStatusMsg = HttpHelper.postFormData(new
Uri("http://www.linkedin.com/groupMsg"), PostMessage);
```

Finally result looks like in LinkedIn send mgs section,



Never-before offers on your favourite Samsung products


You

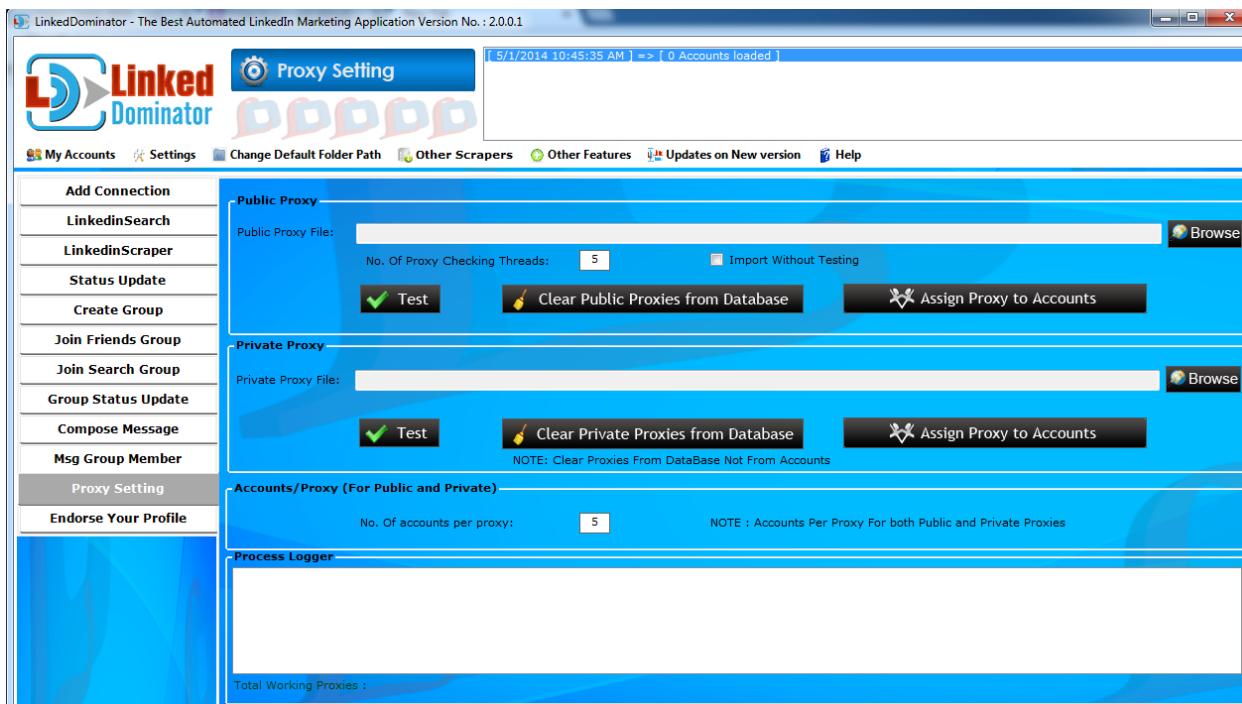
Samsung CPP

Get amazing discounts on the entire Samsung product range

[Start shopping!](#)

### 1.15 Proxy Setting

You want to have proxies with your account or you want to just test them then you can use this feature to test private/public proxies and then you can assign them.



It has two section

**1.15.1 Public Proxy** If you have public proxies with you which you want to use with the software or accounts then write down them in a txt file and upload that in the public proxy section and then press 'Test' button and it will give you a text file with all the working and non-working proxies kept separately in different text files. Clicking on 'Assign Proxy to Account' will assign the working public proxy tp the uploaded accounts in the software.

### 1.15.2 Private proxy

If you have private proxies with you which you want to use with the software or accounts then write down them in a txt file and upload that in the public proxy section and then press 'Test' button and it will give you a text file with all the working and non-working proxies kept separately in different text files.

Clicking on 'Assign Proxy to Account' will assign the working private proxy tp the uploaded accounts in the software.

### How it work with code,

In public proxy, browse public proxy and add lstProxy variable,

```
lstProxy_AccCreator.Clear();
lstProxy_AccCreator.AddRange(lstProxies);
AddToProxysLogs("[ " + DateTime.Now + " ] => [ " + lstProxies.Count + " Public Proxy
Uploaded ]");
```

Test Public proxy list for its working/valid or not its start with single thread manner

```
new Thread(() =>
{
    GetValidProxies(lstProxies);
}).Start();
```

After that its call another method “**ThreadPoolMethod\_Proxies**” with multithreaded manner

```
foreach (string item in lstProxies)
{
    if (!proxy_stop)
    {
        ThreadPool.SetMaxThreads(numberOfProxyThreads, 500);
        ThreadPool.QueueUserWorkItem(new WaitCallback(ThreadPoolMethod_Proxies), new
object[] { item });
    }
}
```

In process with method “**ThreadPoolMethod\_Proxies**” call and work with other method

“**CheckProxy()**”

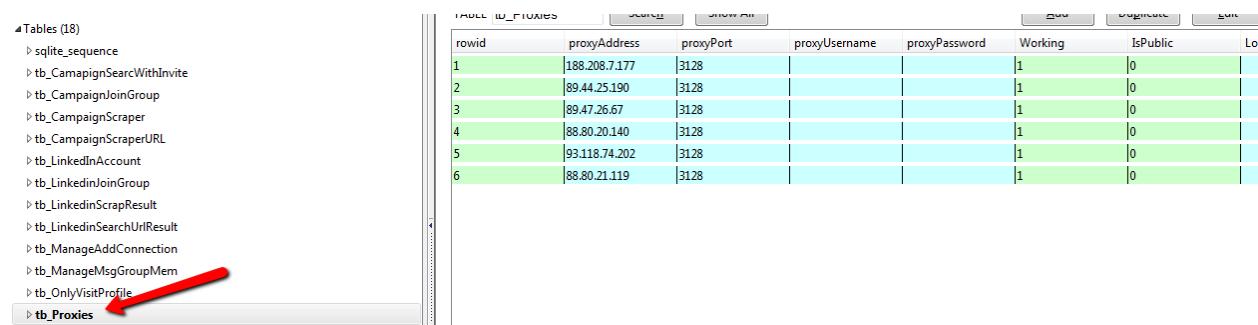
“**CheckProxy**” method is responsible for check public proxy one by one for LinkedIn site while its working or not if get request proper response with same proxy its added working proxy list.

Get request for check proxy

```
string pageSource = HttpHelper.getHtmlFromUrlProxy(new Uri("http://www.linkedin.com/"),
proxyAddress, int.Parse(proxyPort), proxyUsername, proxyPassword);
```

Also working Public proxy is maintained separate table,

```
Working = 1;
string InsertQuery = "Insert into tb_Proxies values('" + proxyAddress + "','" + proxyPort +
"','" + proxyUsername + "','" + proxyPassword + "','" + Working + "','" + IsPublic + "','" +
LoggedInIp + "')";
 DataBaseHandler.InsertQuery(InsertQuery, "tb_Proxies");
 BaseLib.GlobusFileHelper.AppendStringToTextfileNewLine(proxyAddress + ":" + proxyPort +
 ":" + proxyUsername + ":" + proxyPassword, Globals.Path_ExsistingProxies);
```



rowid	proxyAddress	proxyPort	proxyUsername	proxyPassword	Working	IsPublic	LoggedInIp
1	188.208.7.177	3128			1	0	
2	89.44.25.190	3128			1	0	
3	89.47.26.67	3128			1	0	
4	88.80.20.140	3128			1	0	
5	93.118.74.202	3128			1	0	
6	88.80.21.119	3128			1	0	

Assign Public proxy to accounts which is added in account section,

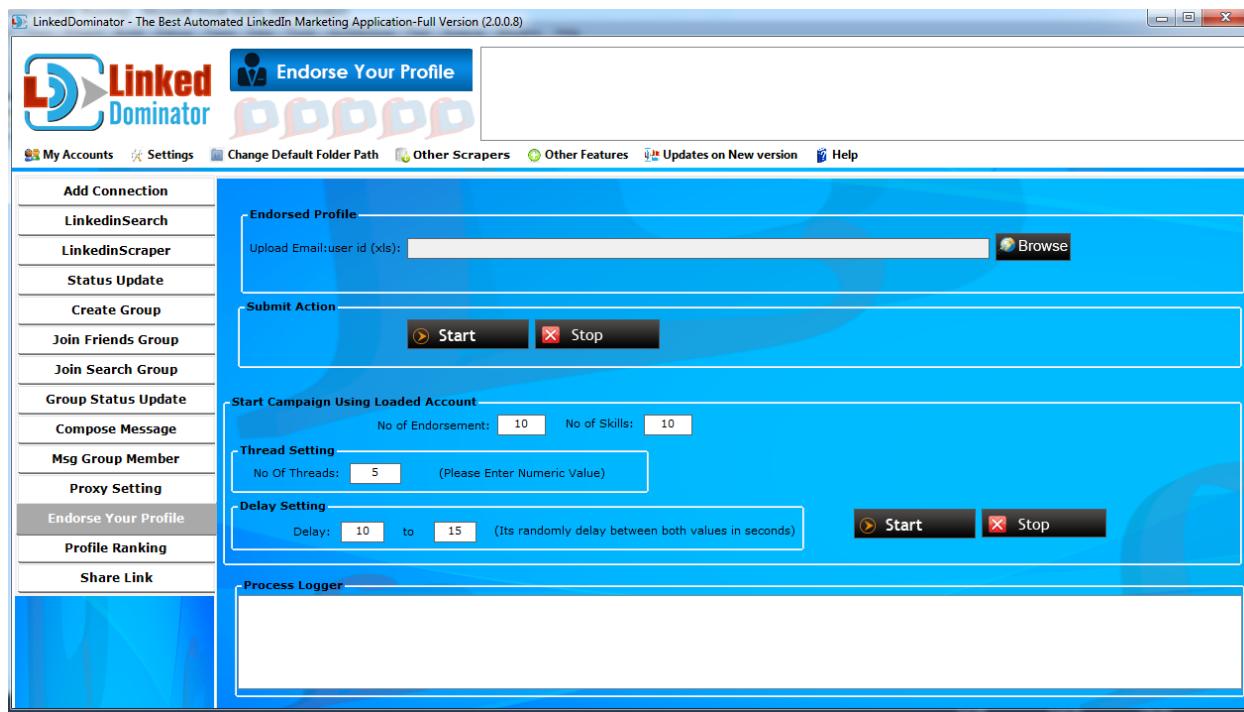
```
List<string> lstProxies = proxyFetcher.GetPublicProxies();
if (lstProxies.Count > 0)
{
    if (!string.IsNullOrEmpty(txtAccountsPerProxy.Text) &&
GlobusRegex.ValidateNumber(txtAccountsPerProxy.Text))
    {
        accountsPerProxy = int.Parse(txtAccountsPerProxy.Text);
    }
}

proxyFetcher.AssignProxiesToAccounts(lstProxies, accountsPerProxy);
ReloadAccountsFromDataBase();
AddToProxysLogs("[ " + DateTime.Now + " ] => [ Proxies Assigned To Accounts ]);
```

Same Logic implemented with Private proxy also.

### 1.16 Endorse Your Profile

Want to have your skills endorsed or you want to endorse someone's skills then you can have this feature and you can endorse their profile.



a) If you want your 1st connection's skill to be endorsed then there is no better tool than this.

b) You just need to get the ID from the profile URL of your 1st connection. Paste that ID in a text file and upload that in the module.

c)Press 'Start' button once you have uploaded the text file and then the software will endorse the skills which It will get in the first hit.

### How it work with code,

First of all browse .xls format file which have endorsed with uploaded account

Normal endorsing Click "Start" Button, its start with single thread with method "**EndorsementPerDay**"

```
Thread _EndorsementPerDay = new Thread(EndorsementPerDay);
_EndorsementPerDay.Start();
```

In process of method "**EndorsementPerDay**" its call another method with multithreaded manner

```
ThreadPool.SetMaxThreads(NoOfThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(EndorsingPerDay), new object[] { item,
NoOfEndorsementPerDay, NoOfSkillsPerDay, MinDelay, MaxDelay });
Thread.Sleep(1000);
```

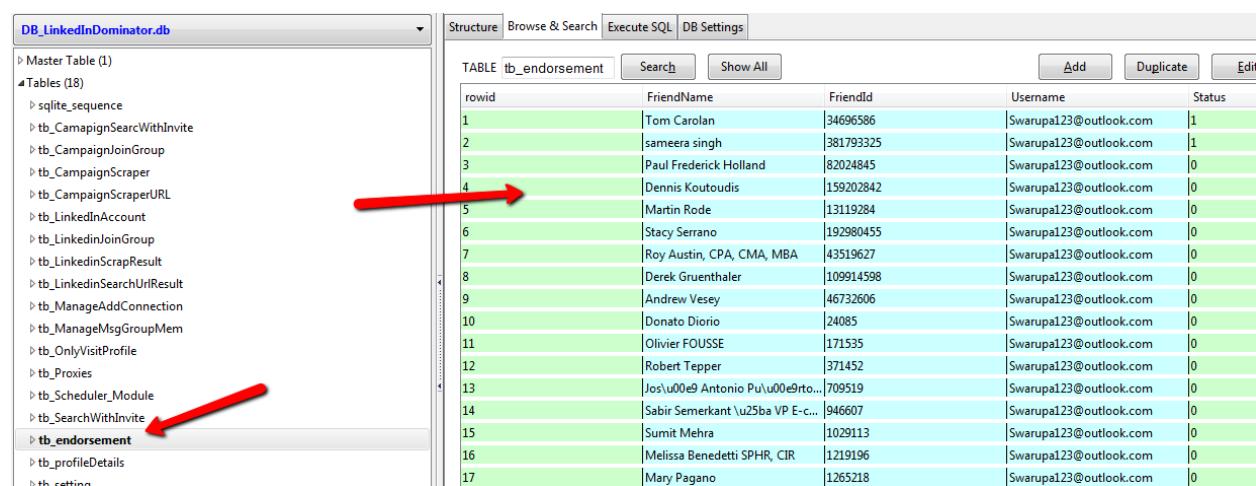
Login process is done with method "**EndorsingPerDay**"

```
Login.LoginHttpHelper(ref HttpHelper);
```

Before scrapping 1<sup>st</sup> members its cross check with db table for data availability,

```
string query = "Select * From tb_endorsement WHERE (Username = '" + item.Key + "')";
DataSet ds = DataBaseHandler.SelectQuery(query, "tb_endorsement");
DataTable dt = ds.Tables["tb_endorsement"];
```

Separate table maintain the record for members,



The screenshot shows a database interface for 'DB\_LinkedInDominator.db'. On the left, a tree view lists tables: Master Table (1), Tables (18), and tb\_endorsement. A red arrow points to the tb\_endorsement entry. On the right, a grid displays the 'tb\_endorsement' table structure with columns: rowid, FriendName, FriendId, Username, and Status. The grid contains 17 rows of data, with a red arrow pointing to the 4th row. The data is as follows:

rowid	FriendName	FriendId	Username	Status
1	Tom Carolan	34696586	Swarupa123@outlook.com	1
2	sameera singh	381793325	Swarupa123@outlook.com	1
3	Paul Frederick Holland	82024845	Swarupa123@outlook.com	0
4	Dennis Koutoudis	159202842	Swarupa123@outlook.com	0
5	Martin Rode	13119284	Swarupa123@outlook.com	0
6	Stacy Serrano	192980455	Swarupa123@outlook.com	0
7	Roy Austin, CPA, CMA, MBA	43519627	Swarupa123@outlook.com	0
8	Derek Gruenthaler	109914598	Swarupa123@outlook.com	0
9	Andrew Vesey	46732606	Swarupa123@outlook.com	0
10	Donato Diorio	24085	Swarupa123@outlook.com	0
11	Olivier FOUSSE	171535	Swarupa123@outlook.com	0
12	Robert Tepper	371452	Swarupa123@outlook.com	0
13	Jos\u00f3n Antonio Pu\u00f3eरtano	709519	Swarupa123@outlook.com	0
14	Sabir Semerkant \u25ba VP E-c...	946607	Swarupa123@outlook.com	0
15	Sumit Mehra	1029113	Swarupa123@outlook.com	0
16	Melissa Benedetti SPHR, CIR	1219196	Swarupa123@outlook.com	0
17	Mary Pagano	1265218	Swarupa123@outlook.com	0

If record is exist in below table its scrap the friends,

```
if (Username != item.Key)
```

```
{
    AddLoggerEndorsePeople("[ " + DateTime.Now + " ] => [ Scraping 1st Connection from
Account : " + item.Key + " ]");
    GroupStatus MemberScrape = new GroupStatus();
    GroupStatus.moduleLog = "endorsecamp";
    Dictionary<string, string> Result = MemberScrape.PostAddMembers(ref HttpHelper,
Login.accountUser);
    AddLoggerEndorsePeople("[ " + DateTime.Now + " ] => [ Scraping process completed
from Account : " + item.Key + " ]");
}
```

In Process of method "**PostAddMembers**" after scrapping the members with Id,  
Insert table which is not found the table,

```
string Query = "INSERT INTO tb_endorsement (FriendName, FriendId, Username, Status)
VALUES ('" + MemFullName + "','" + MemId1 + "','" + user + "','0')";
 DataBaseHandler.InsertQuery(Query, "tb_endorsement");
```

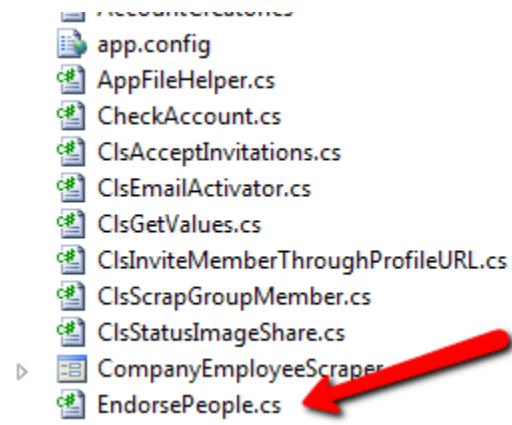
After inserted records its select record which is "0" status means all zero status rows are not endorsed till now,

```
string SelectQuery = "select * from tb_endorsement where Username = '" + item.Key + "'"
and Status=0";
DataSet dst = DataBaseHandler.SelectQuery(SelectQuery, "tb_endorsement");
DataTable dt1 = dst.Tables["tb_endorsement"];
```

After this it's collect all friends id,

```
LstFriendId.Add(FriendId);
```

After that its call another method "**EndorsingPeople**" this method have write with separate class "**EndorsePeople**"



### **EndorsePeople.EndorsingPeople(ref HttpHelper, \_FriendId);**

This method is used final process of endorse skill First of all its get request with friend id after request getting csrf token and endorse url if added only members key.

After that its added key skill for working account finally post request for endorse skill and update db status "0" to "1" and also added csv list for all endorse skill with current working account.

## Final Post request for endorse skills

After endorsed successfully table updated

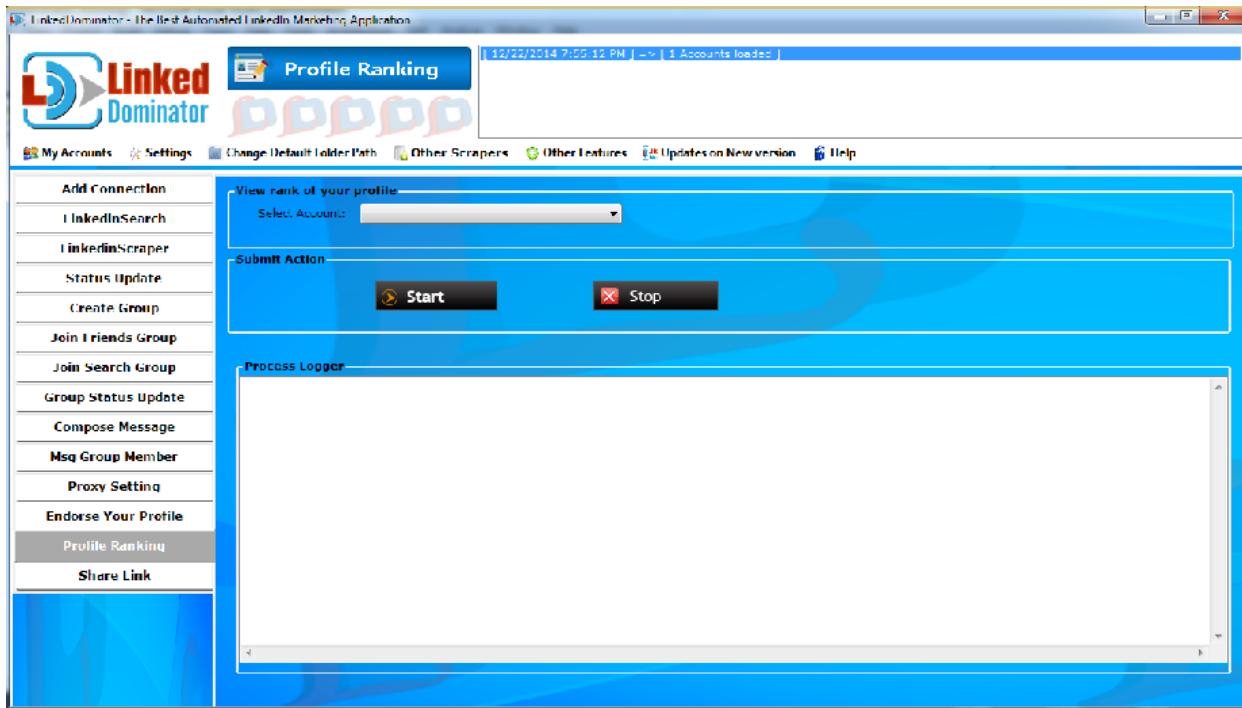
After above post request its looks like in LinkedIn site, profile section  
Also same functionality with campaign extra settings are available in campaign section like  
No of Skill, No of thread with delay setting.

## Skills & Endorsements



## 1.17 Profile Ranking

We fetch you rank of your profile among your connection along with percentile. Also, we fetch name and rank of your top 10 connections. Along with that we give your number of your profile views in total and number of profile views with different keywords used.

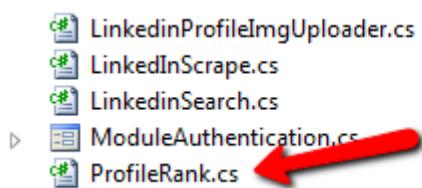


### Working with Code,

Its start with method “[StartProfileRank](#)” its start with single thread

```
Thread thread_ProfileRank = new Thread(StartProfileRank);
thread_ProfileRank.Start();
```

This method “[StartProfileRank\(\)](#)” responsible for account login after successfully login another method run



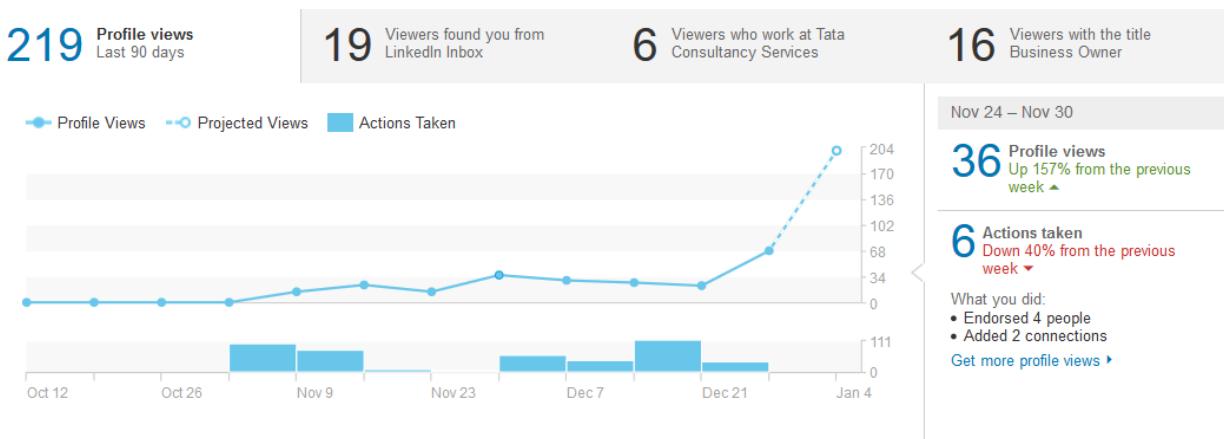
```
obj_profileRank.LinkedInProfilerank(ref objGlobusHttpHelper);
```

This method first fetches self-detail i.e.(name, ID, Image Url, profile percentile on the basis of company, profile rank on the basis of company, profile percentile on the basis of connection, profile rank on the basis of connection) and writes the data in a CSV file with the name SelfRankDetails.csv and to a text file with the file name SelfRankDetails.txt

It also, gives profile view statistics i.e. number of profile views on the basis of occupation, source used to view the profile, company where the profile viewers work at, industry in which profile viewers are, keywords used by the profile viewers to get to the profile, area in which profile viewers live and writes the data in a CSV file with the file name PofileViewStats.csv

And, It gives the top 10 profile ranks of connections as well. It writes the data into a CSV file with the file name MemberRankDetails.csv and to a text file with the file name MemberRankDetails.txt

What we will scrape from LinkedIn Site

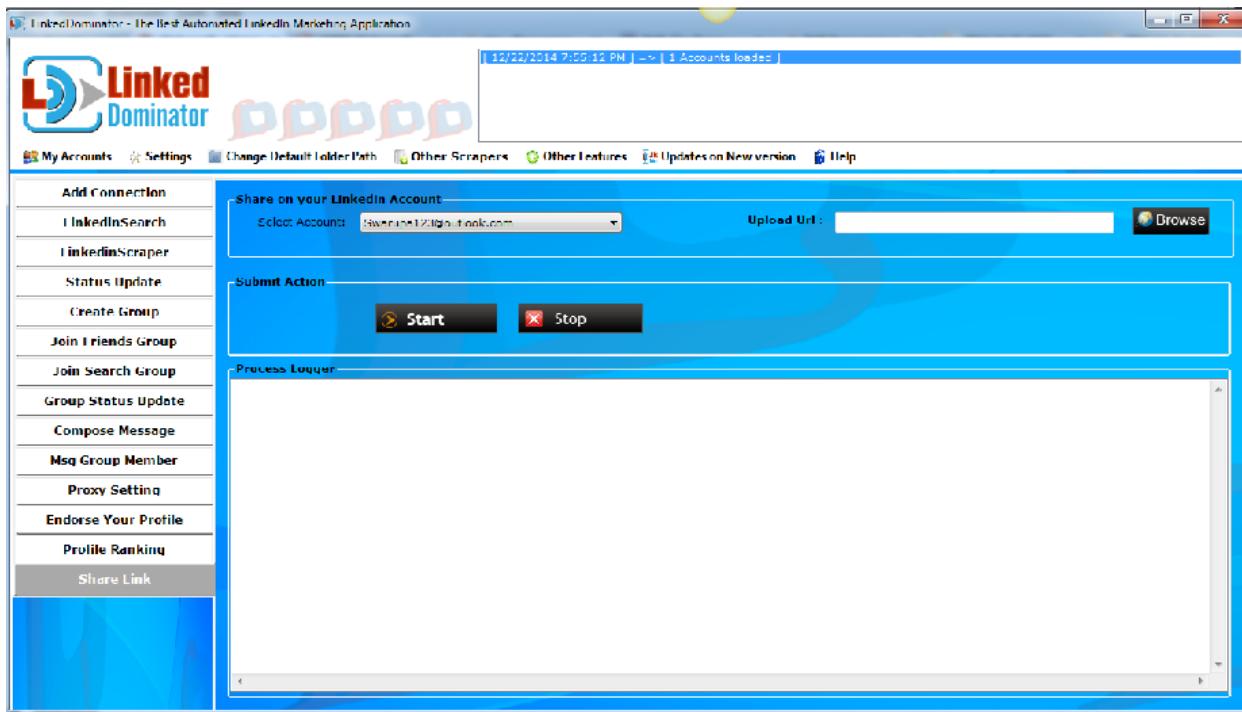


Results looks like in which save csv and text format.

 MemberRankDetails	12/12/2014 3:17 PM	Microsoft Excel C...	9 KB
 MemberRankDetails	12/12/2014 3:17 PM	Text Document	2 KB
 ProfileViewStats	12/12/2014 3:17 PM	Microsoft Excel C...	7 KB
 SelfRankDetails	12/12/2014 3:17 PM	Microsoft Excel C...	1 KB
 SelfRankDetails	12/12/2014 3:17 PM	Text Document	1 KB

### 1.18 Share Url Link

We share the link of the webpage/article/story of the given url (uploaded by user) and display it on the LinkedIn profile's homepage. In case it does not get updated we can check it by going to My Updates.



#### How can work with Code,

```
Thread thread_ProfileRank = new Thread(StartLinkedinShare);
thread_ProfileRank.Start();
```

#### StartLinkedinShare()

This method logs in with the account provided by the user and check for required validations and then if logged in, it picks up each url provided by the user to be shared and passes it to the next method – UrlShare().

```
UrlShare(ref objGlobusHttpHelper, item._Username, item._Password, item._ProxyAddress,
item._ProxyPort, item._ProxyUsername, item._ProxyPassword, webUrlAndShareText);
```

#### “UrlShare()”

This method receives urls uploaded by the user one by one and shares the link of the article/story/webpage on the uploaded LinkedIn profile's homepage.

Final Post Request,

```
string finalPostUrl = "https://www.linkedin.com/sharing/share?trk=LI_BADGE";
```

```
string postDataFinalPost = "csrfToken=" + csrfToken + "&content.id=" + contentId +  
"&content.url=" + Uri.EscapeDataString(contentUrl) + "&content.resolvedUrl=" +  
Uri.EscapeDataString(contentResolvedUrl) + "&contentType=" + contentType +  
"&content.title=" + contentTitle + "&content.description=" + contentDescription +  
"&mentions=" + mentions + "&shareText=" + shareText.Replace(" ", "+") +  
"&dist.networks%5B0%5D=" + distNetwork;  
string responseFinalPost = objGlobusHttpHelper.postFormData(new Uri(finalPostUrl),  
postDataFinalPost);
```

Result looks like in LinkedIn profile home page,



[swarupa rani](#)

[www.globussoft.com](#)

**Welcome to Globussoft**

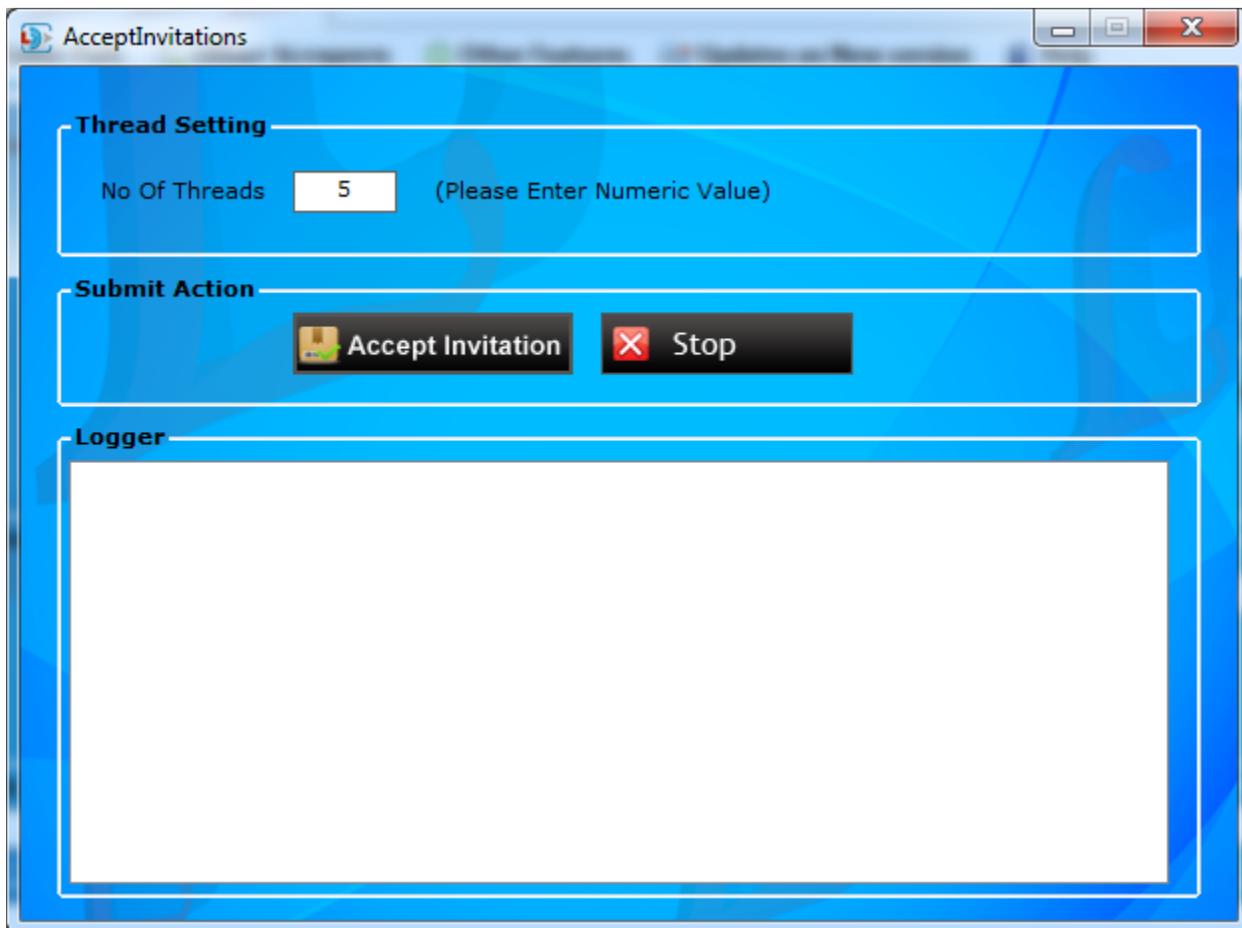
[globussoft.com](#) • Globussoft builds innovative and cutting edge technology products for the digital marketing industry. With over 4500 customers in more than 50 countries across 5 continents and over 2 million users who consume our services...

Like • Comment • Share • 1s ago

## 2. Other Features

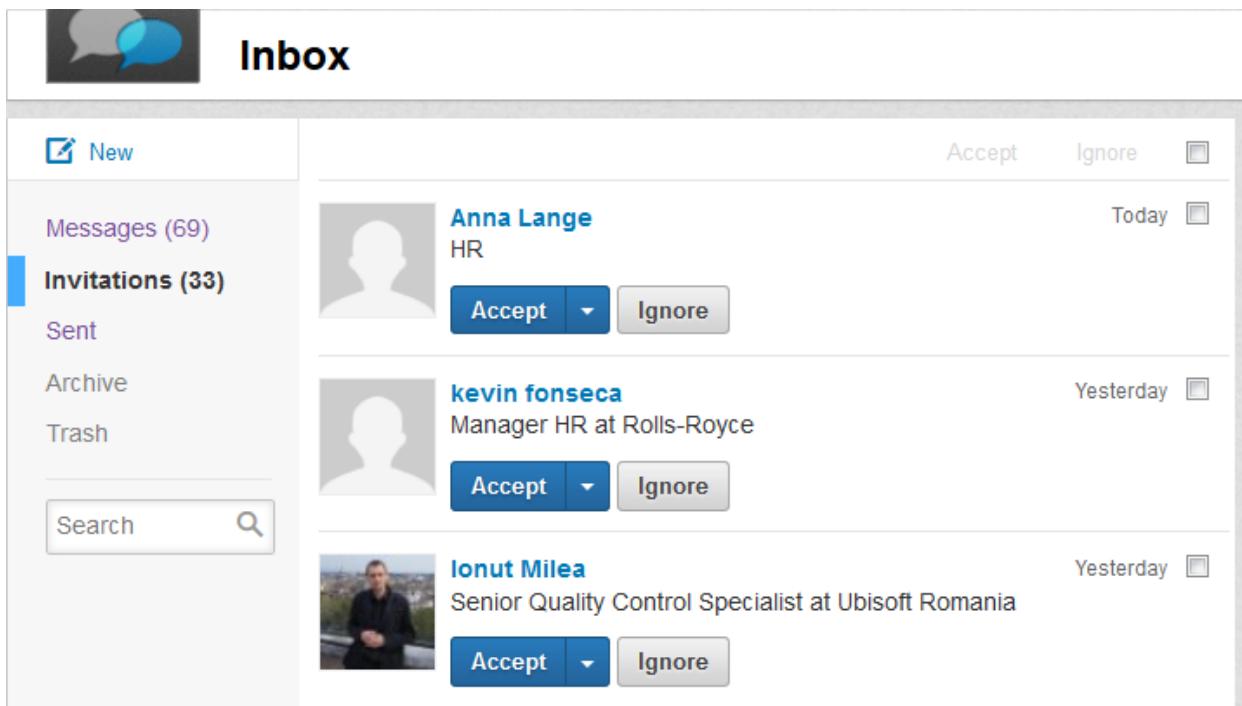
### 2.1 Accept Invitation

Do you have many invitations received in your profiles and its boring to accept everyone one after another. Our module makes it easy for you as it accept all your invitation at a time.



- a) Its very simple to use this module.
- b) You just need to press the "Accept Invitation" button and the software will accept all the invitation present in all the profiles you uploaded.
- c) Once all the invitations are accepted you will get a message indicating that the process is completed for accepting invitation.

**How it work with code,**  
Accept Invitation Inbox,



The screenshot shows the LinkedDominator inbox interface. On the left, there's a sidebar with options like 'New', 'Messages (69)', 'Invitations (33)' (which is selected and highlighted in blue), 'Sent', 'Archive', and 'Trash'. Below that is a search bar. The main area displays three invitation notifications:

- Anna Lange**, HR (Today): Accept, Ignore
- kevin fonseca**, Manager HR at Rolls-Royce (Yesterday): Accept, Ignore
- Ionut Milea**, Senior Quality Control Specialist at Ubisoft Romania (Yesterday): Accept, Ignore

After click "**Accept Invitation**" button its start with single threaded manner

```
Thread acceptInvitation_Thread = new Thread(StartAcceptInvitation);
acceptInvitation_Thread.Start();
```

this method is passes another method

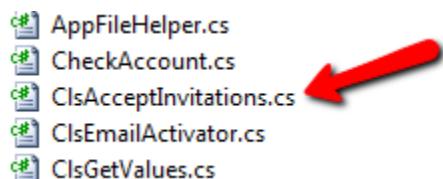
#### "StartAcceptInvitation"

This method is first of all validate no thread and passes another method "**StartAcceptInvitationMultiThread**" with multithreaded manner,

```
ThreadPool.SetMaxThreads(NoofThread, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(StartAcceptInvitationMultiThread), new
object[] { item });
Thread.Sleep(1000);
```

#### "StartAcceptInvitationMultiThread"

Functionality of this method first of all its login of loaded account, logger setting after successfully login its passes other method "**StartAcceptInvitations**"



```
obj_ClsAcceptInvitations.StartAcceptInvitations(ref HttpHelper);
```

This method is start with get request with get all pending request

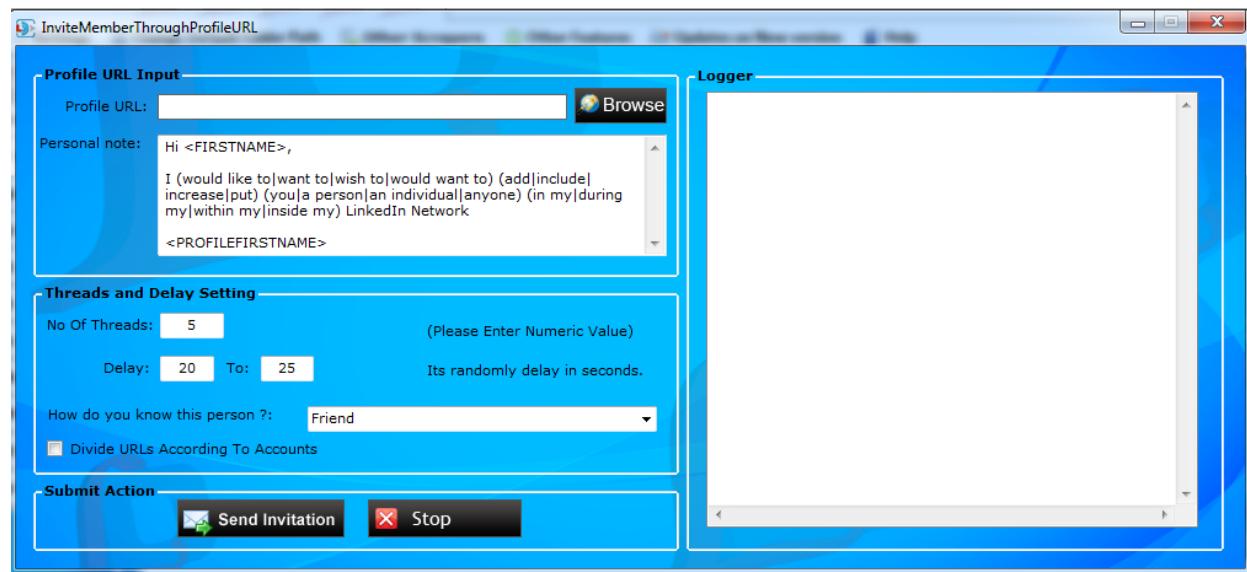
```
string pageSource = httpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/inbox/invitations/pending"));
```

after request its get csrfToken and get another request with get inbox pending request and get sender name and get other get request for accept invitation confirmation. And save the successfully confirmation list in text file.

```
string response2 = httpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/inbox/action?mboxItemGID=" + data_gid +
"&actionType=invitationAccept&csrfToken=" + csrfToken +
"&goback=%2Epiv_*1_*1_*1_*1&trk=inbox-invitations-inv-
accept&ctx=inbox&rnd=1366352095313"));
```

## 2.2 Invite Member through Profile url

Do you have lots of profile urls to whom you want to send invitation then you can upload them all in this module and software will send them invitation to all of them at a once.



- The first step will be to get the url of the profiles.
- Copy and paste the profile url in a text file and save it.
- Now upload that saved text file in profile url section.
- After you have uploaded the required file press 'Send Invitation' button and wait for the process to gets complete.
- Once the process gets complete, you will get a message in the logger sating that the process is complete.

### How it work with Code,

Before start send invitation button, added profile urls using browse option personnel note setting with spintax setting Thread and delay setting personnel profile and divide url according to accounts etc after complete setting

Click "Send Invitation" button

First of all validation all setting and run with method with single threaded manner,

```
Thread acceptInvitation_Thread = new Thread(StartSendInvitation);
acceptInvitation_Thread.Start();
```

After this process passes another method

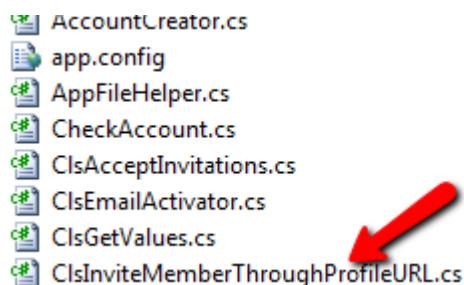
**"StartSendInvitation"**

Its method setting thread with multithread check account url with accounts if check true of option divide url according to accounts after that its passes with multithreaded manner with other method

```
ThreadPool.SetMaxThreads(NoofThread, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(StartSendInvitationMultyThread), new
object[] { item });
```

Method "**StartSendInvitationMultyThread**"

In this method, login account after that login its passes method "**StartSendInvitations**"



```
obj_ClsInviteMemberThroughProfileURL.StartSendInvitations(ref HttpHelper);
```

In method "**StartSendInvitations**" its passes method "**Connect**" with foreach loop

In "**Connect**" method after first get request

```
pageSource = httpHelper.getHtmlFromUrl1(new Uri(profileURL));
```

we will get key, authtoken, goback setting, full name etc all values are necessary for next get request

```
string pageSource1 = httpHelper.getHtmlFromUrl1(new
Uri("http://www.linkedin.com/people/invite?from=profile&key=" + key + "&firstName=" +
firstName_Guest + "&lastName=" + lastName_Guest + "&authToken=" + authToken +
"&authType=name&goback=" + goback + "&trk=prof-0-sb-connect-button"));
```

after this request other method "**SendInvitation**" with all parameters

```
SendInvitation(ref httpHelper, pageSource1, key, authToken, goback, firstName_Guest,
lastName_Guest);
```

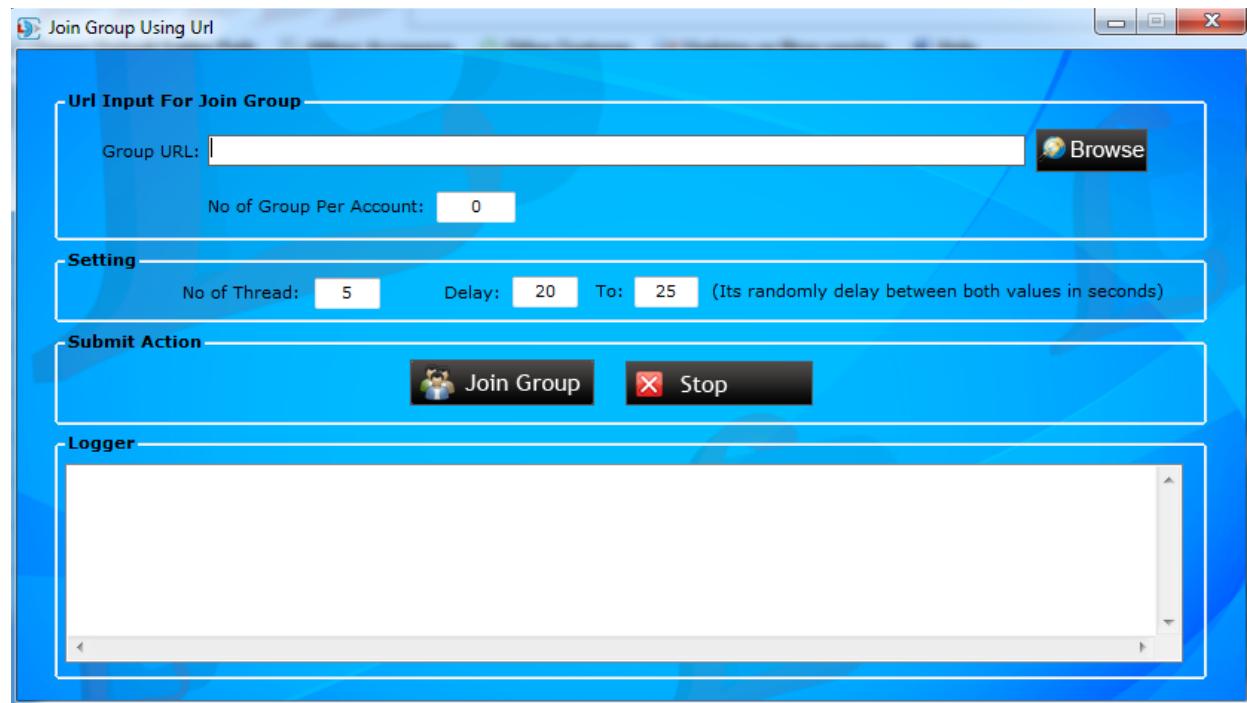
Finally this method get all details for final post request

```
string postData
="existingPositionIC=&companyName.0=&titleIC.0=&startYearIC.0=&endYearIC.0=&school
Text=&schoolID=&existingPositionIB=&companyName.1=&titleIB.0=&startYearIB.0=&endY
earIB.0=&reason=IF&otherEmail=&greeting=" + Uri.EscapeDataString(messagebody) +
"&iweReconnectSubmit=Send+Invitation&key=" + key + "&firstName=" + firstName_Guest
+ "&lastName=" + lastName_Guest + "&authToken=" + authToken +
"&authType=name&trk=prof-0-sb-
connectbutton&iweLimitReached=false&companyID.0=&companyID.1=&schoolID=&schoolc
ountryCode=&schoolprovinceCode=&javascriptEnabled=false&existingAssociation=Job+Ope
nings%2C+Job+Leads+and+Job+Connections%21&subject=" + subject + "&defaultText="
+ defaultText + "&csrfToken=" + csrfToken + "&sourceAlias=" + sourceAlias + "&goback="
+ goback + "";
string postResponse = httpHelper.postFormData(new
Uri("http://www.linkedin.com/people/iweReconnectAction"), postData);
```

After post final request all successfully requested account id save in csv format.

### 2.3 Join Group using URL

Do you have the urls of the groups and you want all the profile to join them then our module will help you with that. It will allow all the accounts loaded to go and join the groups you have uploaded. [www.linkeddominator](http://www.linkeddominator.com).



- a) If you have a list of group urls which you want your list/single account to go and joining in a matter of minutes then you can use this module.
- b) You need to upload the list of group url in the following format. '  
<https://www.linkedin.com/groups?mostRecent=&gid=83347>'
- c) Please ignore the rest of the characters came along with the url, you just need the url till the gid number.
- d) Take the list of urls and paste them in a text file and save them.
- e) Now upload that txt file in 'group url' section and select the 'No of group per account' which will decide that how many groups per account will be added.
- f) Once you are done with it press 'join group' button and wait for the process to get complete. You will get a message of process completion in logger which will indicate that all the accounts which you have uploaded have joined/not joined the groups.

### **How it work with code,**

Before submit action button "Join Group" we have added GroupUrls with accounts Enter No of groups per account, Divide data setting like divide equally or divide given by user, no of thread and delay setting after that we will click "Join Group" Button for start process in this feature.

After click the submit action button it's validated some basic settings after that Its start with single threaded manner with method "[LinkedInAddSearchGroups\(\)](#)"

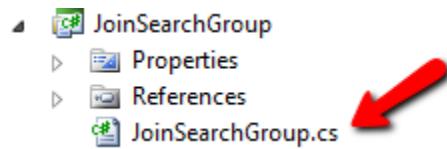
```
new Thread(() =>
{
    LinkedInAddSearchGroups();
}).Start();
```

In this method check account loaded or not divide data setting thread setting after that its passes another method "[PostAddGroupUrl](#)" with multithreaded manner,

```
ThreadPool.QueueUserWorkItem(new WaitCallback(PostAddGroupUrl), new object[] {
item,IstJoinGroupUrl });
index++;
Thread.Sleep(1000);
```

In method "[PostAddGroupUrl](#):

No of thread, delay setting is passes with object based, Login process running  
After successfully login its passes other method "[PostAddOpenGroupsUsingUrl](#)"



```
Dictionary<string, string> Result = obj_JoinSearchGroup.PostAddOpenGroupsUsingUrl(ref
HttpHelper, Login.accountUser, minDelay, maxDelay, IstJoinGroupUrl, IsDevideData);
LinkdInContacts.Add(Login.accountUser, Result);
```

In method “PostAddOpenGroupsUsingUrl”

After first get request

```
string pageSource = HttpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/home?trk=hb_tab_home_top"));
```

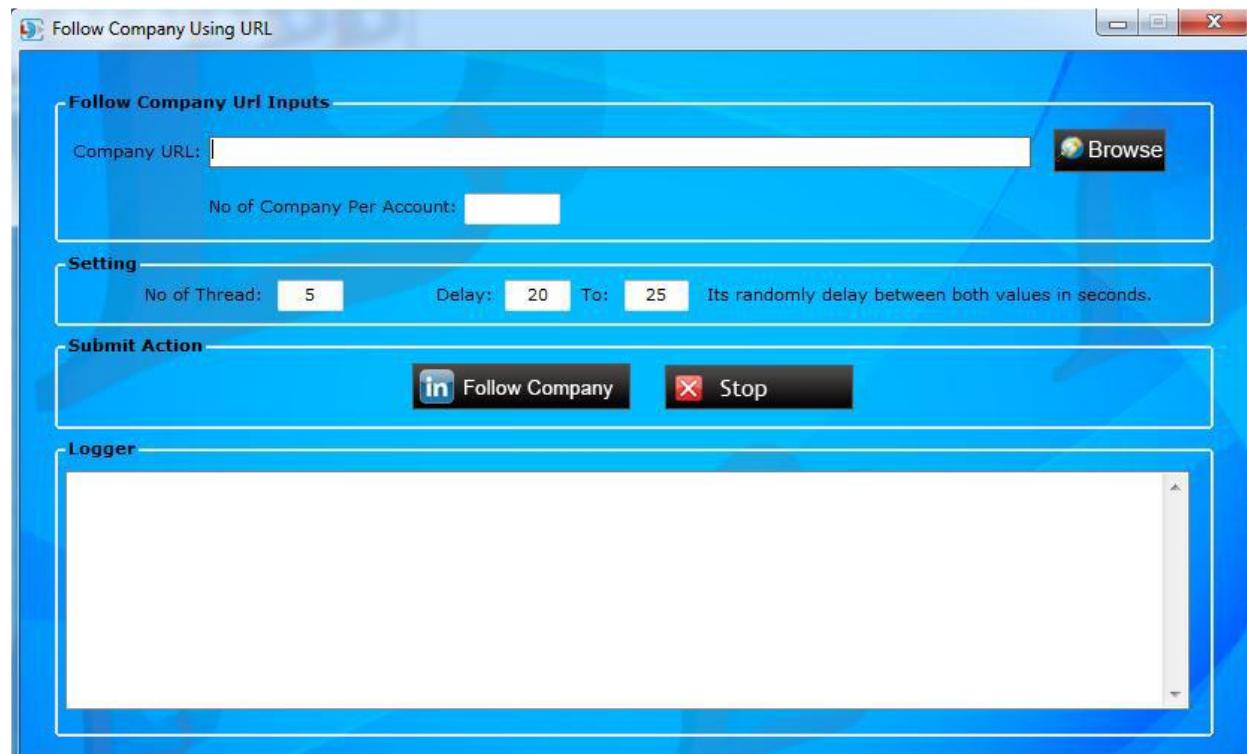
We have getting token, source alias, group name, group id etc,  
After that passes all parameters with final get request,

```
string GoBack = "/%2Eanb_" + GrpId;
LogGroupUrl("[ " + DateTime.Now + " ] => [ ID: " + username + " has Joining the Group:
" + GrpName + " ]");
string pageGetreq = HttpHelper.getHtmlfromUrl1(new
Uri("http://www.linkedin.com/groupRegistration?gid=" + GrpId + "&csrfToken=" +
csrfToken + "&trk=group-join-button")));
```

After successfully joined the group added csv format with all accounts.

## 2.4 Follow Company using URL

You want to follow a company with all the accounts you have then software will help you with that. Just upload the company url and software will go and follow them.



a) If you have a company or you want to increase the follower of a certain/list of companies then in that case you have use this powerful feature present in LinkedDominator to follow them.

b) You just need to copy and paste the company's url in a text file and save it.

c) Go to 'Company url' section an upload the text file where in you saved the company url and select 'No of Company per account' count which will decide that how many companies will be followed by how many accounts.

d) Once you are set press 'Follow Company' button and wait for the process to get complete. You will get a process completion message in the logger which will indicate that accounts have followed the companies you have uploaded.

### **How it work with Code,**

Before click submit action button "Follow Company" add Follow Company url with account, text input setting option for no of company per account, Thread setting and delay setting after that we have click submit button,

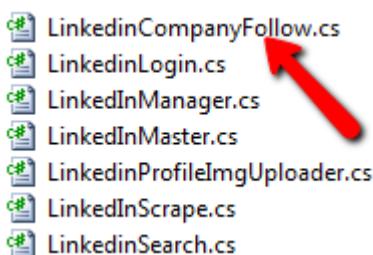
After click submit button, process start with method "[LinkedInAddSearchCompany\(\)](#)" with single threaded manner,

```
new Thread(() =>
{
    LinkedInAddSearchCompany();
}).Start();
```

After process this method its passes another method "[PostAddcompanyUrl](#)" with multithreaded manner,

```
ThreadPool.SetMaxThreads(numberOfThreads, numberOfThreads);
ThreadPool.QueueUserWorkItem(new WaitCallback(PostAddcompanyUrl), new object[] {
    item });
Thread.Sleep(500);
```

After running method its responsible for account Login, Delay setting after that its passes other method "[PostAddCompanyUsingUrl](#)" its manage separate class



LinkedDominator Document Ver 1.0

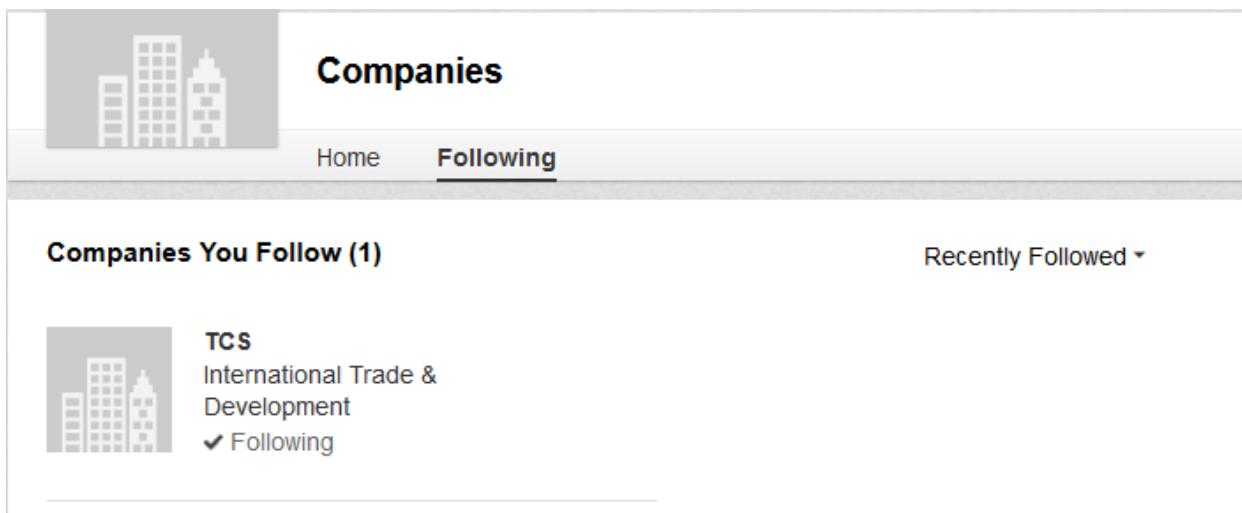
```
Login.LoginHttpHelper(ref HttpHelper);
```

```
string MessagePosted = obj_FollowCompany.PostAddCompanyUsingUrl(ref HttpHelper,  
Login.accountUser,minDelay, maxDelay);
```

Start processing with method “**PostAddCompanyUsingUrl**” first get request for getting csrf token, sourceAlias, company name, company id after getting all value finally it get request for follow the company,

```
string pageGetreq = HttpHelper.getHtmlFromUrl1(new  
Uri("http://www.linkedin.com/company/follow/submit?id=" + CompanyId +  
&fl=start&version=2&ft=pageKey%3Dbiz-overview%3Bmodule%3Dbutton&sp=biz-  
overview&csrfToken=" + csrfToken + "&goback=" + goback + "&ajax=&rnd=" + txid));
```

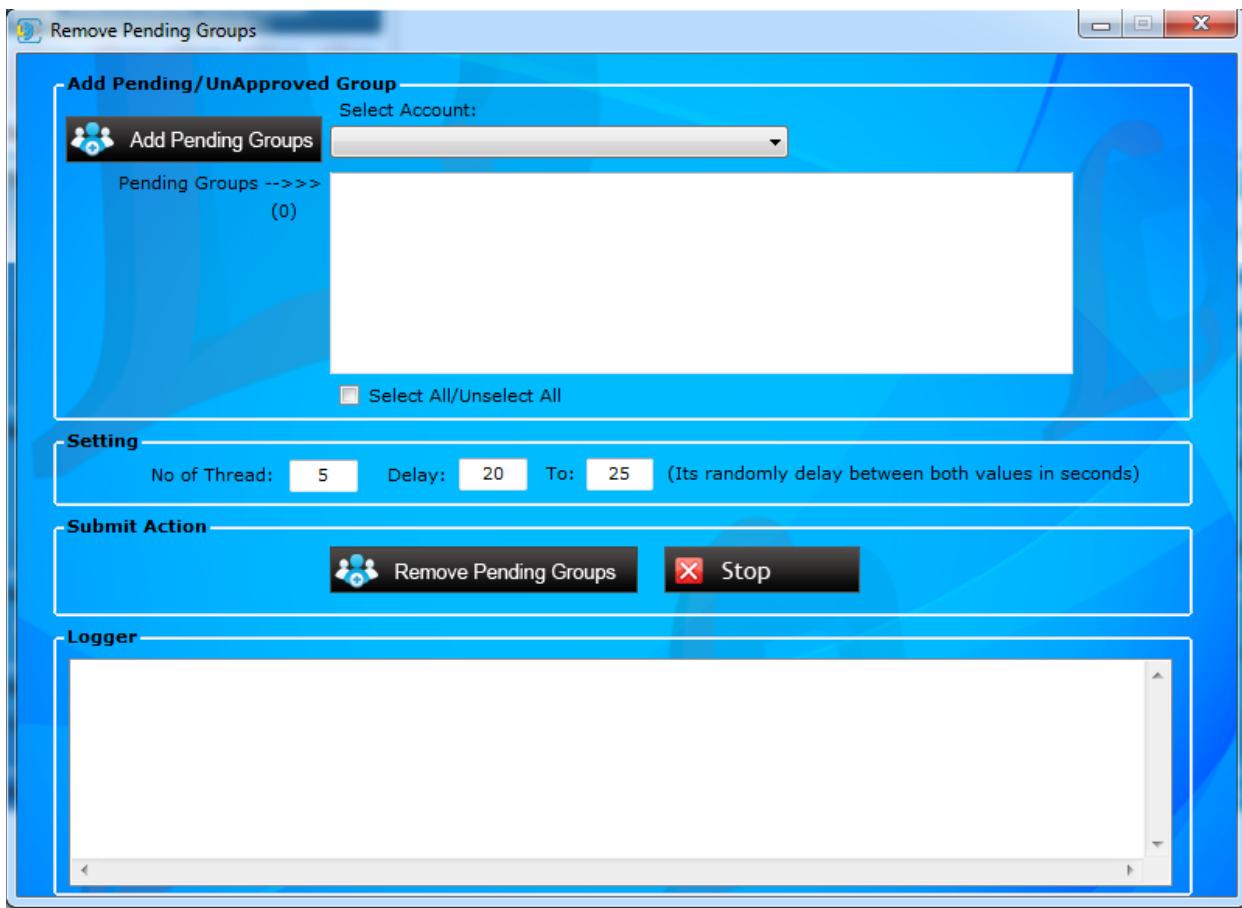
After successfully follow its maintain follow id with account in csv format.



The screenshot shows a LinkedIn interface for managing followed companies. At the top, there's a navigation bar with a city icon and the word "Companies". Below the navigation bar, there are two tabs: "Home" and "Following", with "Following" being underlined. The main content area has a heading "Companies You Follow (1)". Below this, there is a card for "TCS International Trade & Development". The card includes the company's logo (a stylized cityscape), the acronym "TCS", the company name, and the status "Following" with a checkmark. To the right of the card, there is a link "Recently Followed ▾".

## 2.5 Remove Pending urls

Joined many groups and many of them have not granted you access to the group then with the help of this module you can unfollow all of them at a time.



- a) You might have joined many groups for which your access is in waiting status. And due to this pending status you are not allowed to join more groups.
- b) Click on 'Add Pending Groups' and wait for the software to fetch the list of pending groups.
- c) Once you get the groups select the particular account from the dropdown whose pending groups you want to remove.
- d) Now select the pending group to which you want to remove from the account.
- e) After selecting the particular group/ groups, press 'Remove Pending Groups' and wait for the process to get complete.
- f) Once the process gets complete, you will get a message in the logger saying that the pending status group has been removed and along with that you will get a file in the output folder containing the details of the name of the group along with the corresponding accounts.

#### **How it work with Code,**

Before click submit action button "Remove Groups" first of all add pending group which have not accepted request till after collecting pending group with account wise, select

pending groups its display all pending groups in List box check group which one remove also thread setting and delay setting available after that click submit action button "Remove Groups"

Method "[LinkedInPendingGroupSearch](#)" run with single threaded manner

```
new Thread(() =>
{
    LinkedInPendingGroupSearch();
}).Start();
```

Method "[LinkedInPendingGroupSearch](#)" run and passes another method "[StartMultiThreadedPendingGroupAdd](#)" with multithreaded manner

```
ThreadPool.SetMaxThreads(numberofThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(StartMultiThreadedPendingGroupAdd),
new object[] { item });
Thread.Sleep(1000);
```

This method run with Login process after login its passes another method for scrape pending groups and add all pending group in list box with account wise.

```
Dictionary<string, string> Groups = dataScrape.GetSelectedIDForPendingGroups(ref
HttpHelper, Login.accountUser);
LinkdInContacts.Add(Login.accountUser, Groups);
```

After that click "Remove Groups" Button its work with single threaded with method "[LinkedInRemovePendingGroups](#)"

```
new Thread(() =>
{
    LinkedInRemovePendingGroups();
}).Start();
```

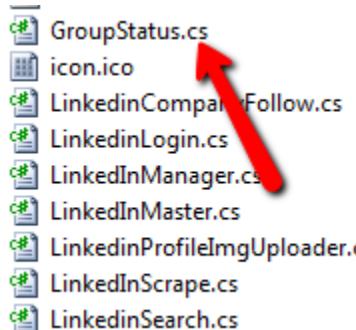
In this method after setting its passes other method "[PostRemovePendingGroups](#)"

```
ThreadPool.SetMaxThreads(numberOfThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(PostRemovePendingGroups), new
object[] { item });
Thread.Sleep(1000);
```

In process of method "[PostRemovePendingGroups](#)" Login process is run

```
Login.LoginHttpHelper(ref HttpHelper);
```

After login its passes other method "[PostRemovePendingGroups](#)" for final processing its write with separate class "Group Status"

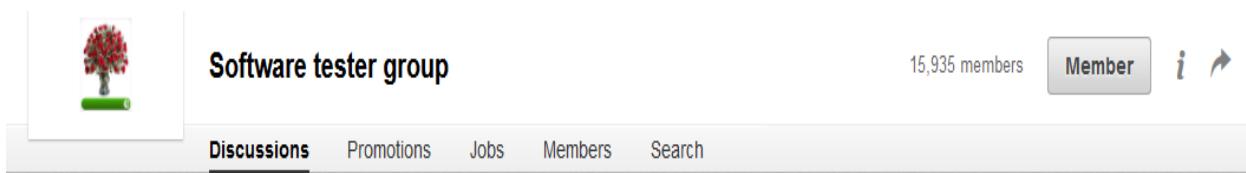


```
string MessagePosted = obj_GroupStatus.PostRemovePendingGroups(Login.accountUser,
Login.accountPass, minDelay, maxDelay);
```

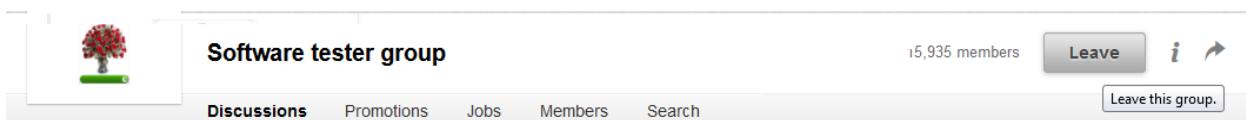
In this method after first get request its getting csrf token, source alias and some necessary post data parameters after that post final get request

```
PostGroupstatus = "http://www.linkedin.com/anet?withdrawJoinRequest=&gid=" +
SelectedGrp.Split(':')[2];
ResponseStatusMsg = HttpHelper.getHtmlfromUrl1(new Uri(PostGroupstatus));
```

Default group looks like in site,

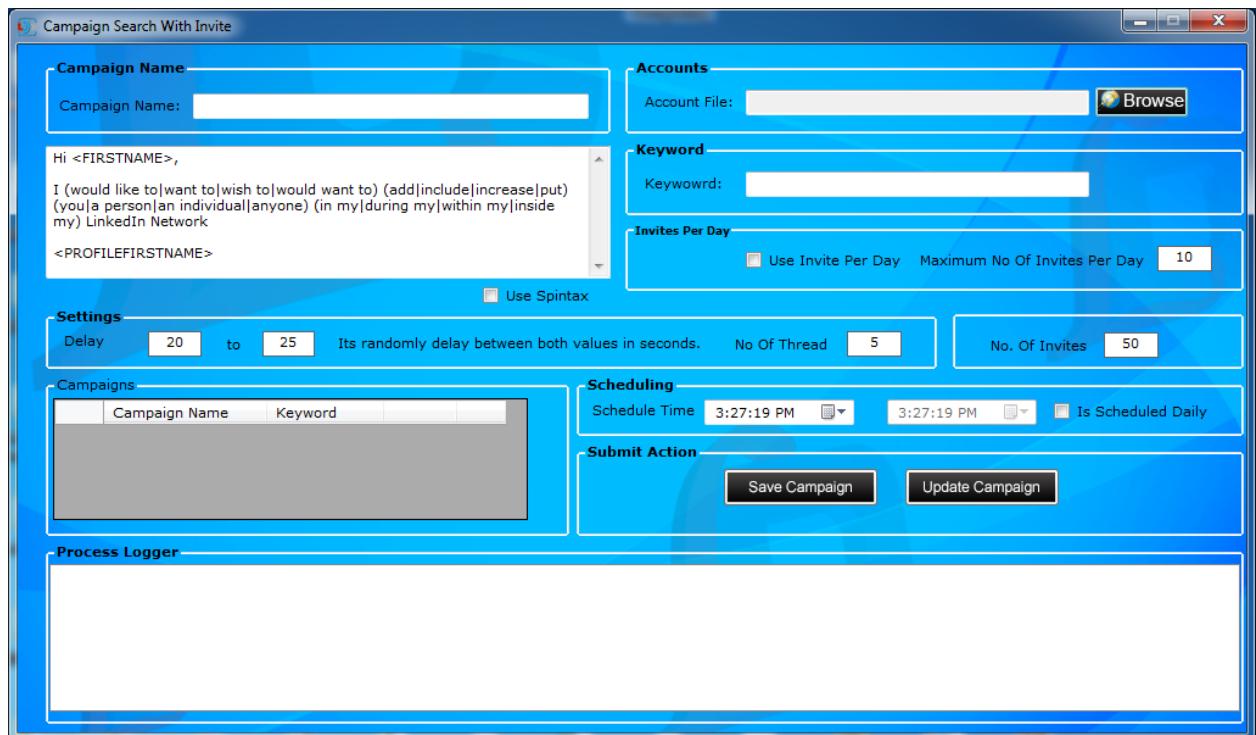


After move the mouse over button if click the "Leave" button its permanent remove for user list,



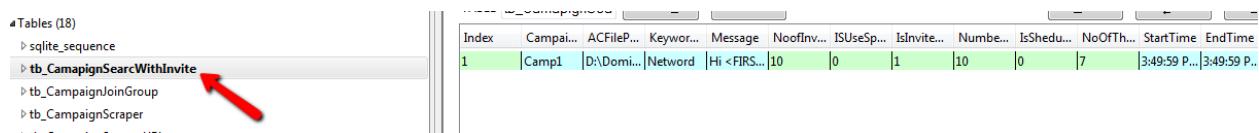
## 2.6 Campaign Invite with Search

This feature is basically Invite connection with search campaign bulk account loaded, invitation mgs with search keyword, invite per day setting, delay setting, no of invites setting, daily scheduling setting after all save campaign with campaign name. Every campaign have single instance more than one campaign run at a time in this feature.



### How it work with Code,

Before save campaign add inputs and setting will be done after that Its passes method “**SaveFollowSettings**” functionality of this method saved all details of campaign in separate table,



Index	CampaignName	ACFileP...	Keyword	Message	NoofInv...	ISUseSp...	IsInvite...	Number...	IsShed...	NoOfTh...	StartTime	EndTime
1	Camp1	D:\Domi...	Network	Hi <FIRSTNAME>,...	10	0	1	10	0	7	3:49:59 PM	3:49:59 PM

```
clsDBQueryManager queryManager = new clsDBQueryManager();
queryManager.InsertCamapaignData(query, "tb_CamapignSearcWithInvite");
```

After that its display and showing with grid view,

Campaigns				
	Campaign Name	Keyword		
▶	Camp1	Netword	<a href="#">Edit</a>	

Above showing green button it responsible for start the campaign

There are two events in greed view

1. Cell Click event
2. Cell Content Click

After click “dgv\_campaign\_CellClick” in greed view there are two options

With image display 1. Img == “ON” (Display green button) means its ready to start campaign 2. Img == “OFF” (Display red button) means its process running mode. Also available here edit campaign means change setting of existing campaign.

There are three option work with this event,

1. `if (e.ColumnIndex == 2)`
2. `else if (e.ColumnIndex == 3)`
3. `else if (Img == "OFF")`

1. `if (e.ColumnIndex == 2)`

If click “Edit” event its passes method “editCampaign” this method is responsible for edit campaign,

```
new Thread(() =>
{
    editCampaign(CampaignName, FeaturName);
}).Start();
```

2. `else if (e.ColumnIndex == 3)`

In click this event with on mode (green button) its start the campaign with method “StartCampaign”

```
new Thread(() =>
{
    StartCampaign(CampaignName, FeaturName);
}).Start();
```

After start this method its added last working thread, scheduling setting after that its passes another method “StartProcess”

```
new Thread(() =>
```

```
{
    StartProcess(CompaignsDataSet, CampaignName);
}

}).Start();
```

After some setting like set campaign details it's memorized with appropriate variables, set thread pool, locker setting after that its start another method  
**"GetStartProcessForFollow"**

```
Thread threadGetStartProcessForFollow = new Thread(GetStartProcessForFollow);
```

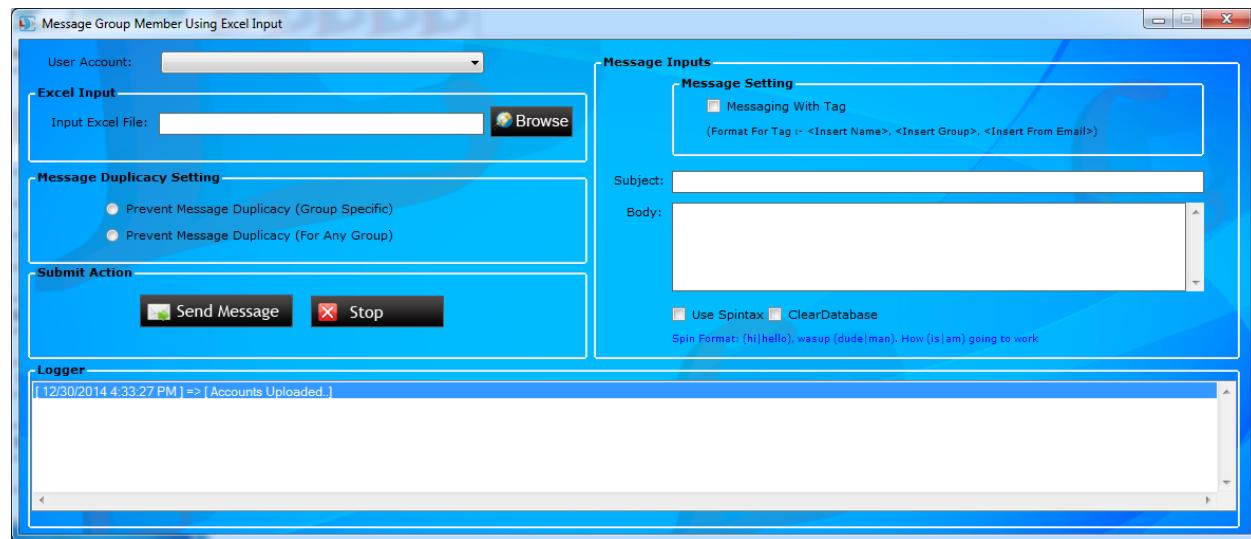
After that its start another method "**SearchUsingkeywordForInvite**"

```
ManageConnections.ConnectUsingSearchKeywod ConnectUsing_Search = new
ConnectUsingSearchKeywod(keyValuePair.Value.Username, keyValuePair.Value.Password,
keyValuePair.Value.proxyAddress, keyValuePair.Value.proxyPort,
keyValuePair.Value.proxyUsername, keyValuePair.Value.proxyPassword,
que_SearchKeywords);
LinkedIn_Master1.SearchUsingkeywordForInvite(ref ConnectUsing_Search, DelayStart,
DelayEnd);
```

This is the final method for connection with search keyword campaign module.

## 2.7 Mgs group member with excel input

This Module same as mgs group member only difference its work with customised excel input files.

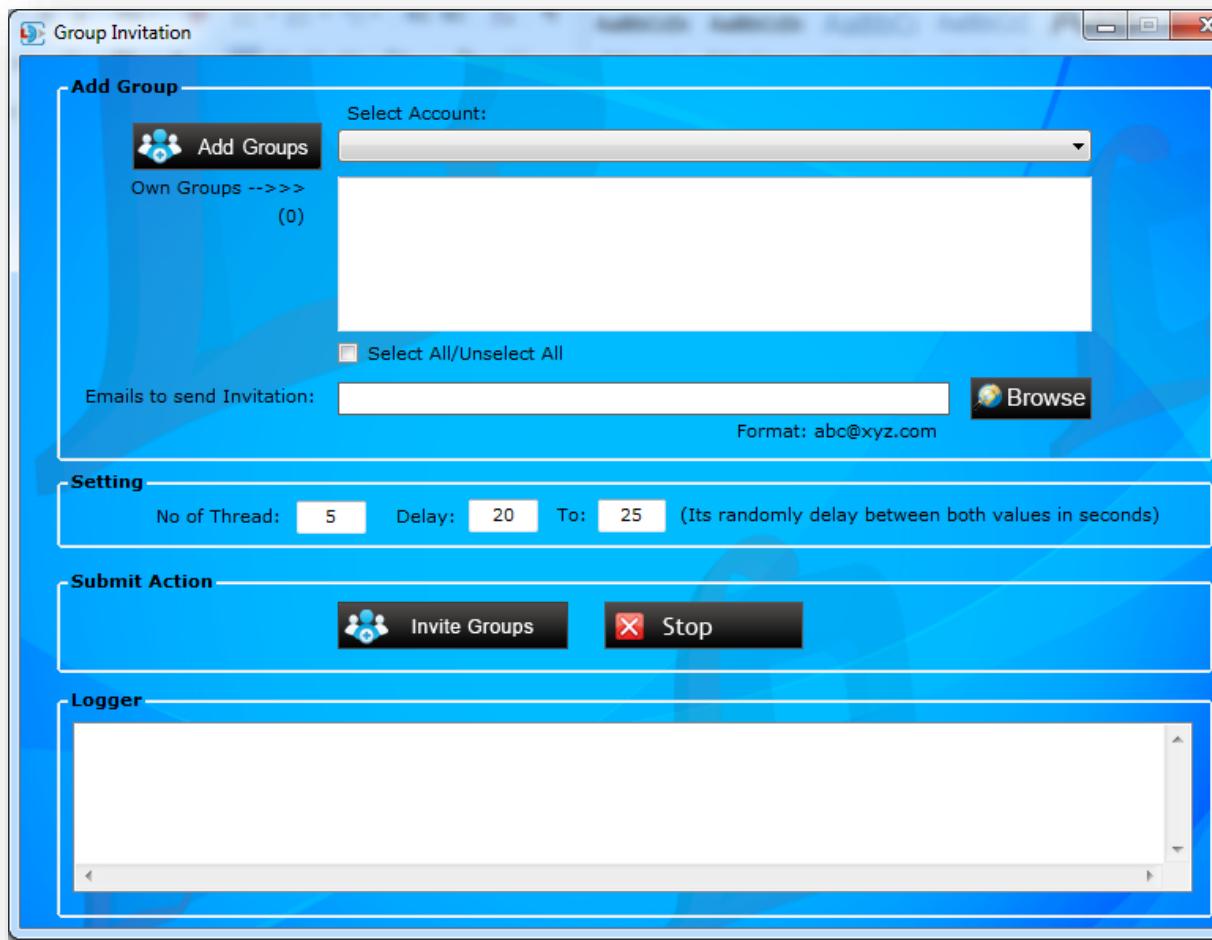


### How can work with code,

Please refer mgs group member describe previously.

## 2.8 Invite Members to Groups

The basic functionality of this module is invitation of other user of LinkedIn in your own group. With email wise if you have add groups with user wise and add email of other LinkedIn user its send mail for join the selected group invitation.



### How it work with code,

If Click "Add Groups" Button its start with method "[LinkedInAddOwnGroup\(\)](#)" with singlw thread

```
new Thread(() =>
{
    LinkedInAddOwnGroup();
}).Start();
```

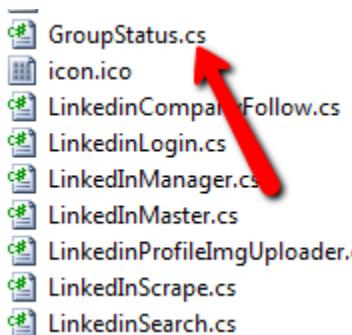
After that its start new method "[StartMultiThreadedOwnGroupAdd](#)" with multithreaded manner

```
ThreadPool.SetMaxThreads(numberofThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(StartMultiThreadedOwnGroupAdd), new
object[] { item });
Thread.Sleep(1000);
```

After login

```
Login.LoginHttpHelper(ref HttpHelper);
```

Its run new method "**GetSelectedIDForOwnGroups**" for group scrapping its written with separate class "**GroupStatus**"



```
Dictionary<string, string> Groups = dataScrape.GetSelectedIDForOwnGroups(ref
HttpHelper, Login.accountUser);
```

After get request

```
pageSourceforGroup = HttpHelper.getHtmlfromUrl1(new
Uri("https://www.linkedin.com/anet?dispSortAnets=&trk=my_groups-h_gn-settings"));
```

Its collect all groups related by selected user.

Before click submit action button "Invite Groups" add emails to send group invitation for selected group which is added by user based,

After that click "Invite Groups" button its start with method "**LinkedInGroupsInvitation**" With single thread

```
new Thread(() =>
{
    LinkedInGroupsInvitation();
}).Start();
```

After run this method its run other method with multithreaded manner with account wise

```
ThreadPool.SetMaxThreads(numberOfThreads, 5);
ThreadPool.QueueUserWorkItem(new WaitCallback(PostGroupsInvitationMail), new object[]
{ item });
Thread.Sleep(1000);
```

Method “**PostGroupsInvitationMail**” start with login process

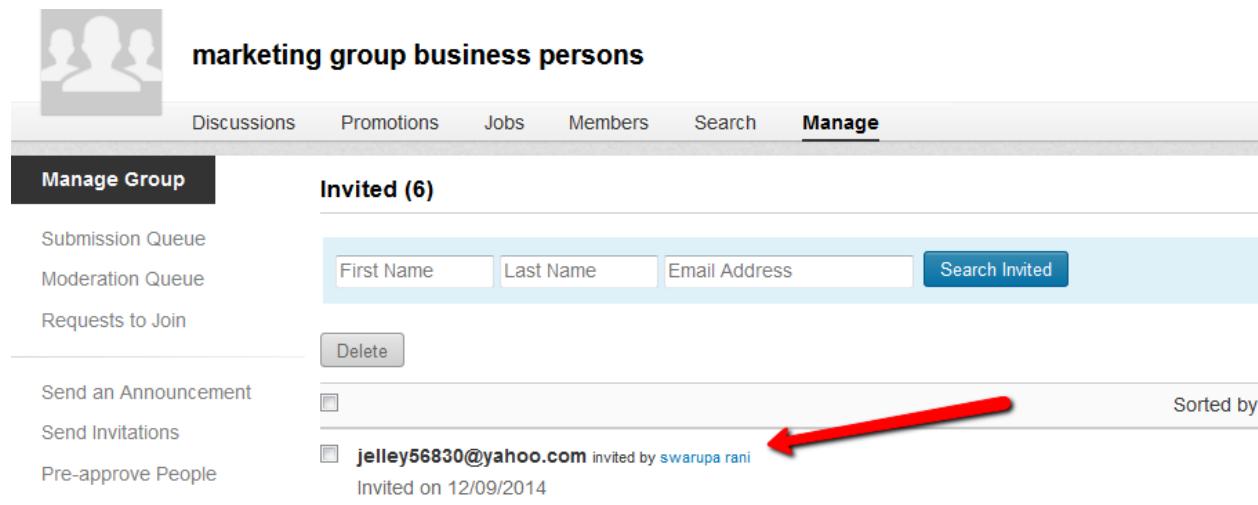
```
Login.LoginHttpHelper(ref HttpHelper);
```

After successfully login with delay setting its run with new method “**PostInvitationGroups**”

```
string MessagePosted = obj_GroupStatus.PostInvitationGroups(Login.accountUser,
Login.accountPass, minDelay, maxDelay);
```

In this method get all parameters like csrfToken, sourcealias, regCsrfParam and some necessary get request its run final post request for send group invitation method

```
string postData = "csrfToken=" + csrfToken + "&emailRecipients=" +
itemEmail.Replace("@", "%40") + "&subAddMbrs=Send+Invitations&gid=" +
SelectedGrp.Split(':')[2] + "&invtActn=im-invite&cntactSrc=cs-connections&remIntives=" +
reminvite + "&connectionIds=&connectionNames=&contactIDs=&newGroup=false";
PostGroupInvite = HttpHelper.postFormData(new
Uri("https://www.linkedin.com/manageGroup"), postData);
```



The screenshot shows the LinkedIn 'marketing group business persons' Manage Group page. The 'Manage' tab is selected. On the left, there's a sidebar with options like 'Manage Group', 'Submission Queue', 'Moderation Queue', 'Requests to Join', 'Send an Announcement', 'Send Invitations', and 'Pre-approve People'. The main area is titled 'Invited (6)' and contains a search bar with fields for 'First Name', 'Last Name', 'Email Address', and a 'Search Invited' button. Below the search bar is a 'Delete' button. A list of invited users is shown, with a red arrow pointing to the email address 'jolley56830@yahoo.com' which was invited by 'swarupa rani' on '12/09/2014'. The list is sorted by date.















