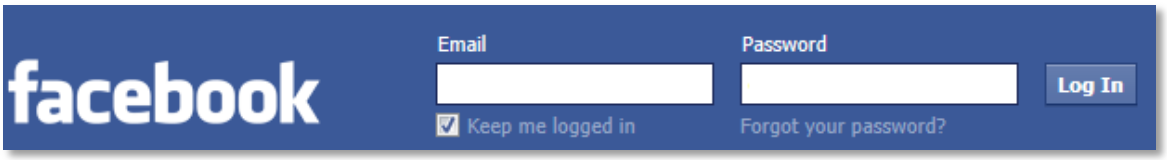
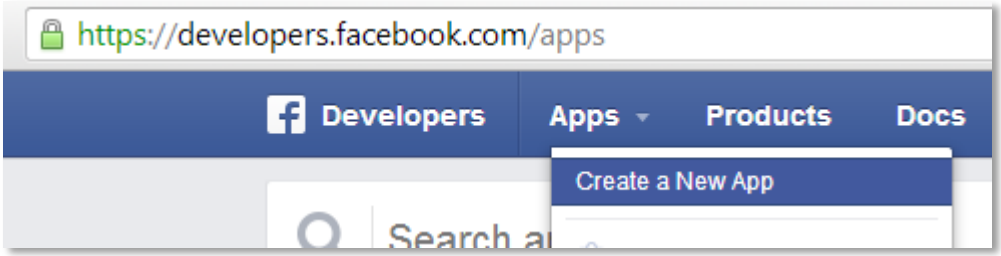
Creating a Facebook desktop application using Facebook SDK and FbGraphApiWrapper

(.NET Framework 4, WinForms, Visual Studio)

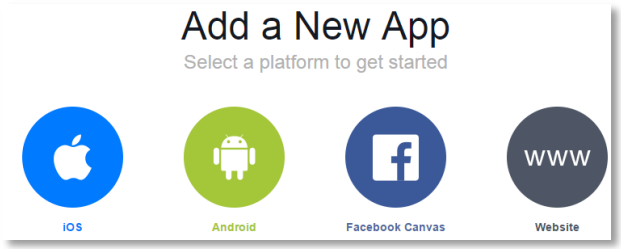
1. **Creating an application-account in Facebook**
   In order to create an application that communicates with Facebook and acts on behalf of your user, you must first create an application-entity in Facebook:
   a. You must have a Facebook account in order to create an application account which will be



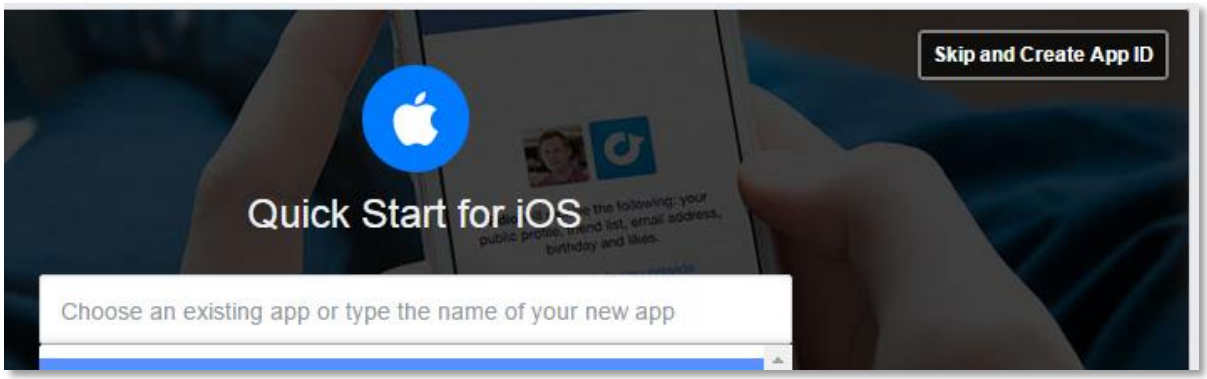
   created under your Facebook account.
   b. Create an Application Account:
      i. Go to https://developers.facebook.com/apps
      ii. Create a new Application Account



      iii. Choose either **iOS** or **Android**



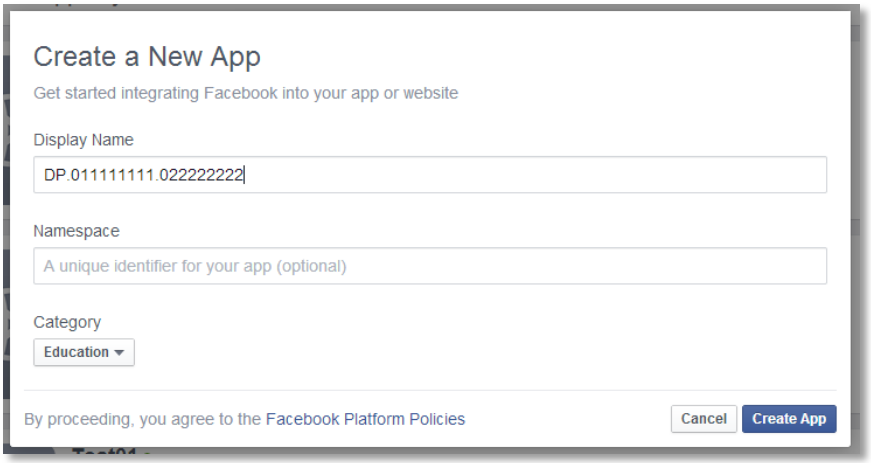
      iv. On the top-right corner, hit the **'Skip and create App ID'** button:



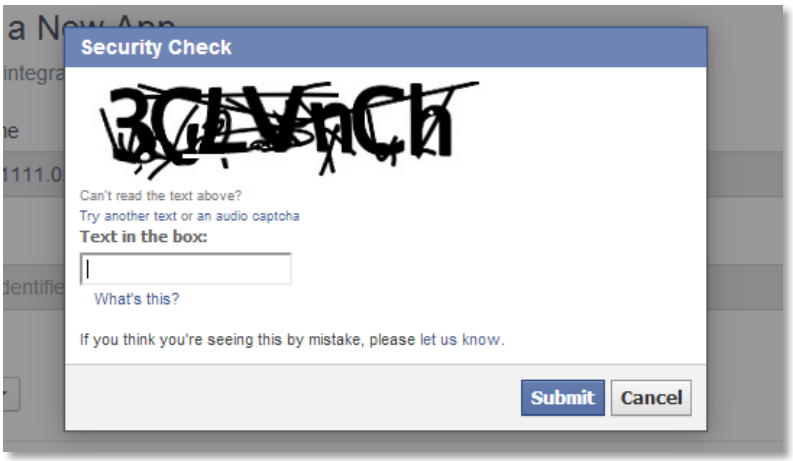
v. Give the application a name using the following format:

DP.011111111.022222222

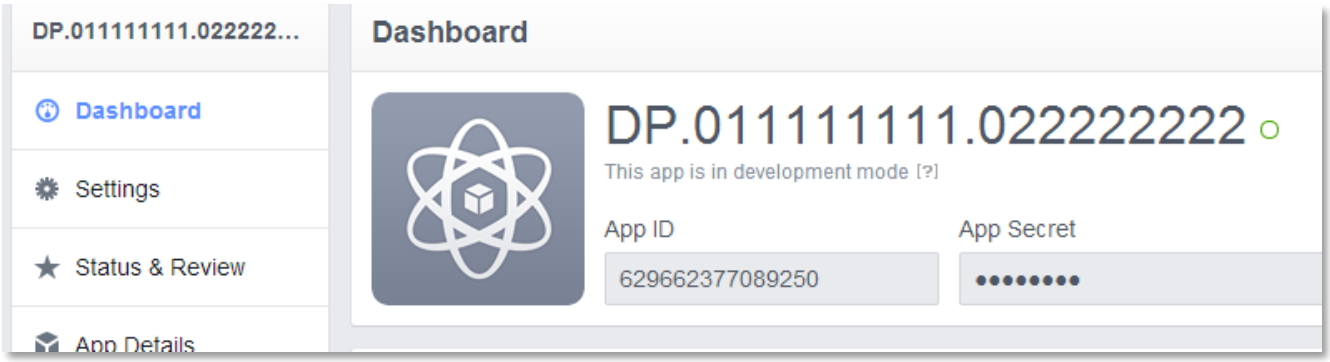
replace 011111111 with the first student's ID number (9 digits!)
replace 022222222 with the second student's ID number (9 digits!)



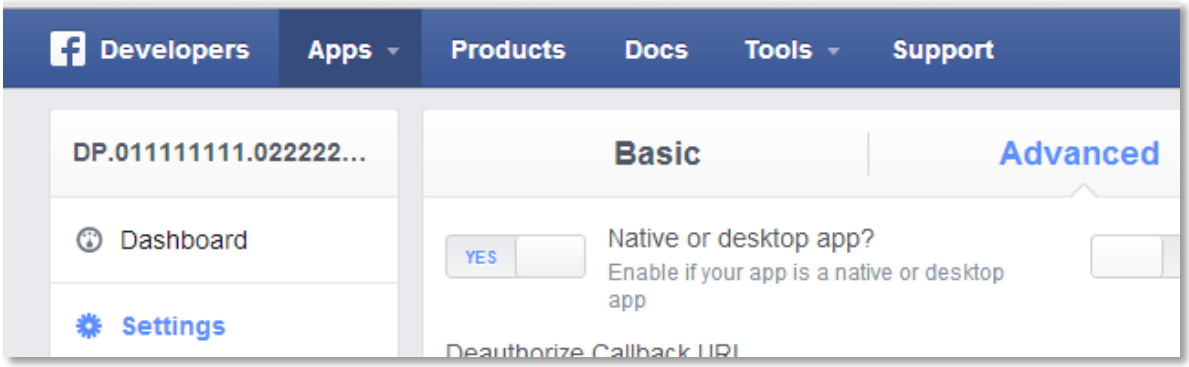
vi. Enter the captcha:



c. Save your newly created App ID for later and then go to 'Settings':

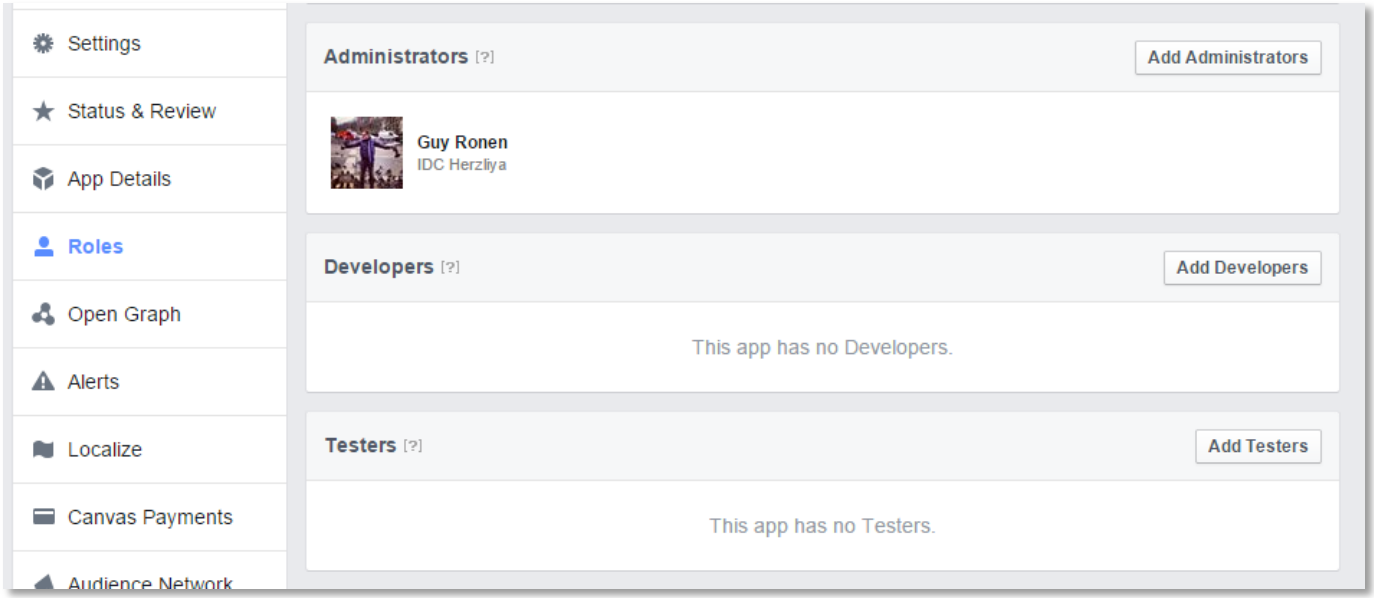


i. In the **'Advanced'** section, **select the Native/Desktop** application type

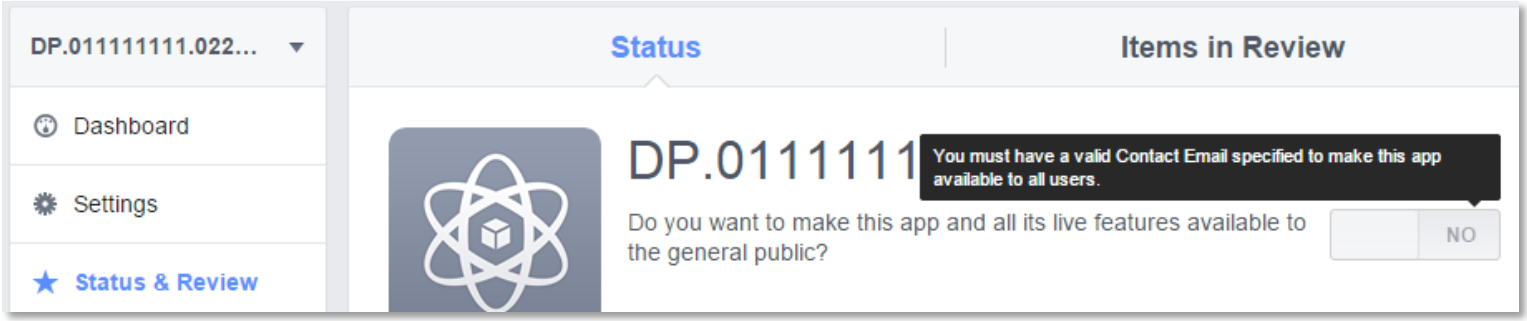


ii. Further down in the Settings/Advanced page, turn on the 'Embedded browser OAuth Login option:

iii. Hit "Save Changed" and go to the 'Roles' tab:

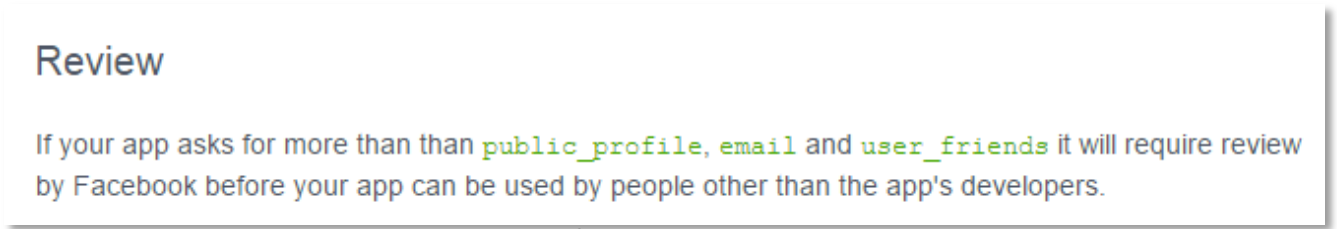


"Roles" is where you can define the "Administrators", "Developers" and "Testers" of your app. Only these users can login through your app until you'll make it available for the general public use:



Add any facebook account to the developers/testers list which you want to be able to test your app. **Specifically, add the http://www.facebook.com/design.patterns user as a tester** (you'll need to be 'friends' with him in order to add him as tester).

NOTE: Even after specifying a 'Contact Email' (in the Settings/Basic section) and turning on the 'available to the general public' switch, you'll still need to send for facebook's review some of the permissions you wish to require from your users, and until they'll be approved, you won't be able to request them from the general public. Only developers/testers will be able to test the app with the yet-to-be-approved permissions:



d. In Visual Studio, create a new **.NET Framework 4.0** WinForms project.

e. From the reference folder of your project, add a reference to the .dll files included in the .zip file of the exercise (FbGraphApiWrapper.dll, Facebook.dll).

f. Use the static login method
`LoginResult result = FacebookWrapper.FBService.Login("272862089537667",`
providing your AppID and the permissions required from your app's user to display a login form to your user.
If this is the first time your user (a facebook account owner) is using your app, he/she will be prompted to approve the permissions requested by your application.
For the list of permission, see this link.

g. The return value of the Login method (LoginResult) has a LoggedInUser property (of type FacebookWrapper.ObjectModel.User) which you should use in order to utilize your user's data and actions, in an object-oriented fashion, for example:

   i. Data:
   user.FirstName, user.LastName, user.Birthday, user.RelationshipStatus, etc.

   ii. Relations to facebook objects:
   user.Friends, user.FriendLists, user.Checkins, user.WallPosts, user.Events, user.Albums, user.Pokes, user.Videos, etc.
   friend.FirstName, friend.LastName, friend.Albums, friend.Checkins, etc.
   album.Photos, checkin.Comments, photo.Comments, photo.Tagged, photo.LikedBy, etc.

   iii. Actions:
   user.PostStatus(), user.PostPhoto(), user.CreateAlbum(), user.CreateFriendList(), etc.
   album.UploadPhoto(), photo.Comment(), photo.Like(), status.Comment(), etc.

h. If the user failed to login or simply closed/canceled the login dialog, the result object will indicate the error with the ErrorMessage property of the LoginResult object.

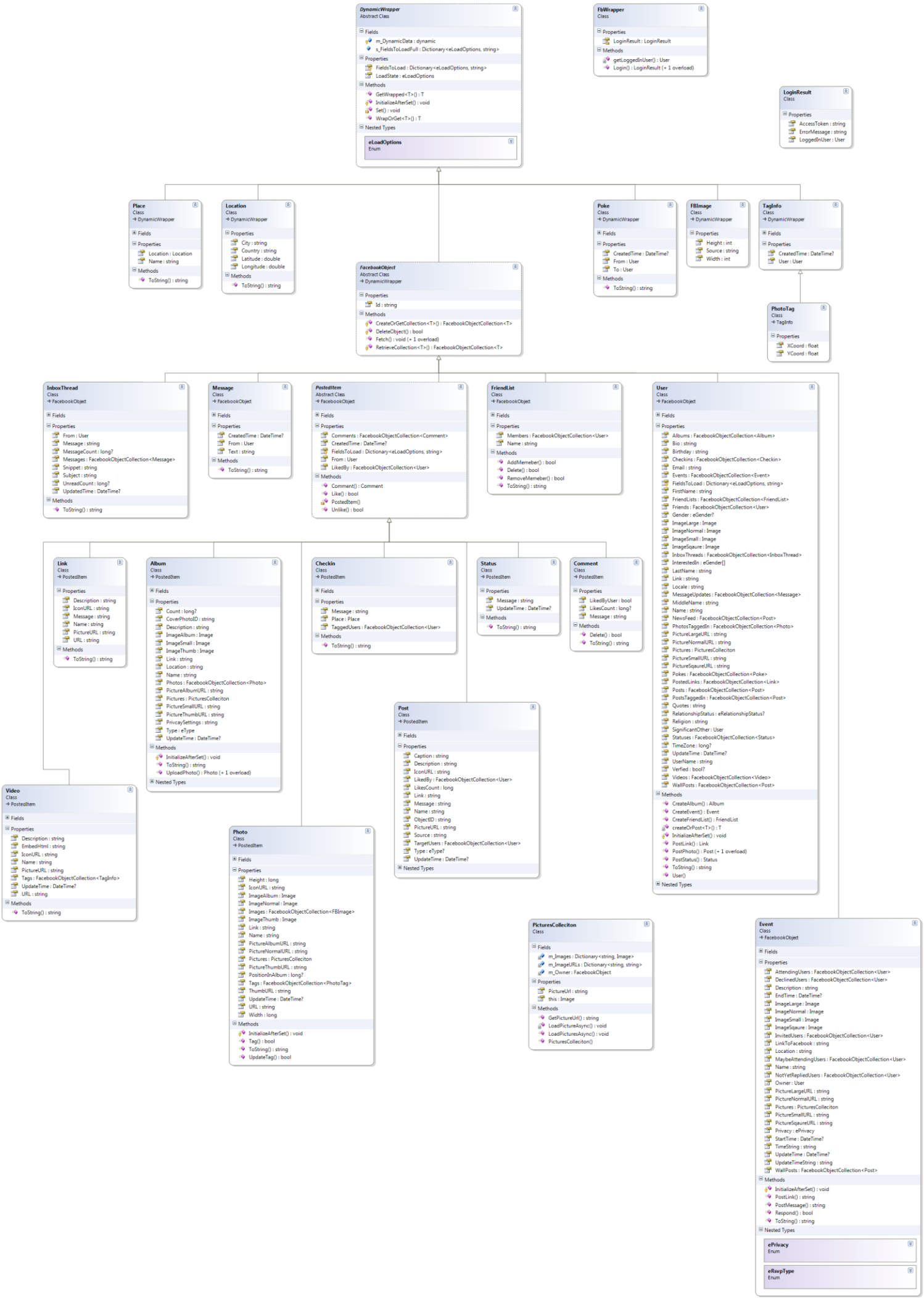
2. Resources:

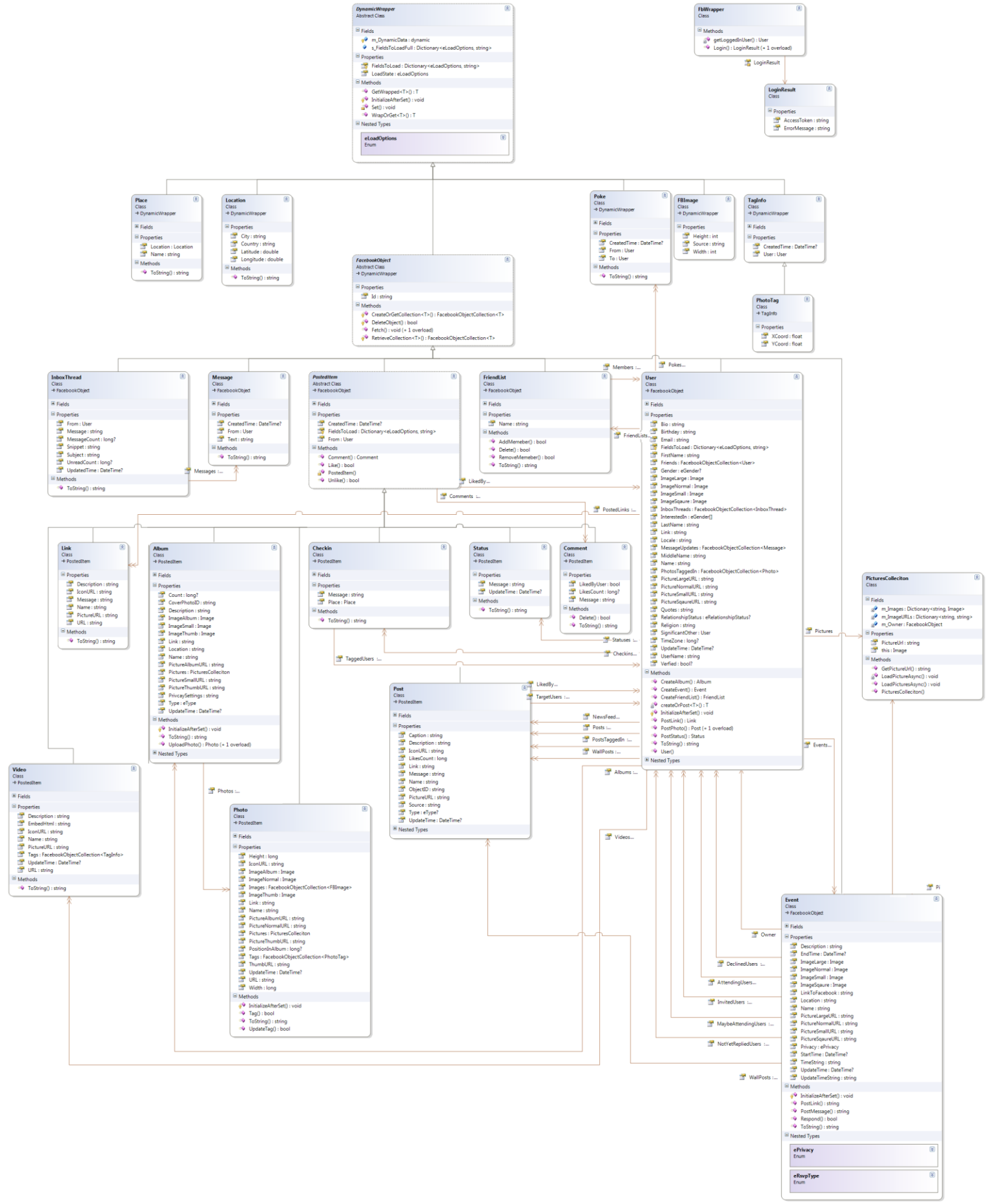
   a. Visit https://developers.facebook.com/docs/reference/api/ to understand more and get all the information about the Facebook Graph API

   b. Use the https://developers.facebook.com/tools/explorer application to browse data on facebook using the Graph API and understanding Jason

   c. The .zip file contains Class Diagrams of the object-oriented wrapper API (.png image files and .cd files which should be viewed in Visual Studio). They are also here on the next pages.
   Use them to learn more about the structure of the API (note: These class diagrams are not complete)

   d. The **ReleaseNote - READ ME!!.txt** file contains interesting information regarding the changes made throughout the different versions of the API. You may find this information useful


DynamicWrapper
Abstract Class
Fields
m_DynamicData : dynamic
s_FieldsToLoadFull : Dictionary<eLoadOptions, string>
Properties
FieldsToLoad : Dictionary<eLoadOptions, string>
LoadState : eLoadOptions
Methods
GetWrapped<T>() : T
InitializeAfterSet() : void
Set() : void
WrapOrGet<T>() : T
Nested Types
eLoadOptions
Enum
FbWrapper
Class
Properties
LoginResult : LoginResult
Methods
getLoggedInUser() : User
Login() : LoginResult (+ 1 overload)
LoginResult
Class
Properties
AccessToken : string
ErrorMessage : string
LoggedInUser : User
Place
Class
→ DynamicWrapper
Fields
Properties
Location : Location
Name : string
Methods
ToString() : string
Location
Class
→ DynamicWrapper
Properties
City : string
Country : string
Latitude : double
Longitude : double
Methods
ToString() : string
Poke
Class
→ DynamicWrapper
Fields
Properties
CreatedTime : DateTime?
From : User
To : User
Methods
ToString() : string
FBImage
Class
→ DynamicWrapper
Properties
Height : int
Source : string
Width : int
TagInfo
Class
→ DynamicWrapper
Fields
Properties
CreatedTime : DateTime?
User : User
FacebookObject
Abstract Class
→ DynamicWrapper
Properties
Id : string
Methods
CreateOrGetCollection<T>() : FacebookObjectCollection<T>
DeleteObject() : bool
Fetch() : void (+ 1 overload)
RetrieveCollection<T>() : FacebookObjectCollection<T>
PhotoTag
Class
→ TagInfo
Properties
XCoord : float
YCoord : float
InboxThread
Class
→ FacebookObject
Fields
Properties
From : User
Message : string
MessageCount : long?
Messages : FacebookObjectCollection<Message>
Snippet : string
Subject : string
UnreadCount : long?
UpdatedTime : DateTime?
Methods
ToString() : string
Message
Class
→ FacebookObject
Fields
Properties
CreatedTime : DateTime?
From : User
Text : string
Methods
ToString() : string
PostedItem
Abstract Class
→ FacebookObject
Fields
Properties
Comments : FacebookObjectCollection<Comment>
CreatedTime : DateTime?
FieldsToLoad : Dictionary<eLoadOptions, string>
From : User
LikedBy : FacebookObjectCollection<User>
Methods
Comment() : Comment
Like() : bool
PostedItem()
Unlike() : bool
FriendList
Class
→ FacebookObject
Fields
Properties
Members : FacebookObjectCollection<User>
Name : string
Methods
AddMemeber() : bool
Delete() : bool
RemoveMemeber() : bool
ToString() : string
User
Class
→ FacebookObject
Fields
Properties
Albums : FacebookObjectCollection<Album>
Bio : string
Birthday : string
Checkins : FacebookObjectCollection<Checkin>
Email : string
Events : FacebookObjectCollection<Event>
FieldsToLoad : Dictionary<eLoadOptions, string>
FirstName : string
FriendLists : FacebookObjectCollection<FriendList>
Friends : FacebookObjectCollection<User>
Gender : eGender?
ImageLarge : Image
ImageNormal : Image
ImageSmall : Image
ImageSquare : Image
InboxThreads : FacebookObjectCollection<InboxThread>
InterestedIn : eGender[]
LastName : string
Link : string
Locale : string
MessageUpdates : FacebookObjectCollection<Message>
MiddleName : string
Name : string
NewsFeed : FacebookObjectCollection<Post>
PhotosTaggedIn : FacebookObjectCollection<Photo>
PictureLargeURL : string
PictureNormalURL : string
Pictures : PicturesCollection
PictureSmallURL : string
PictureSquareURL : string
Pokes : FacebookObjectCollection<Poke>
PostedLinks : FacebookObjectCollection<Link>
Posts : FacebookObjectCollection<Post>
PostsTaggedIn : FacebookObjectCollection<Post>
Quotes : string
RelationshipStatus : eRelationshipStatus?
Religion : string
SignificantOther : User
Statuses : FacebookObjectCollection<Status>
TimeZone : long?
UpdateTime : DateTime?
UserName : string
Verfied : bool?
Videos : FacebookObjectCollection<Video>
WallPosts : FacebookObjectCollection<Post>
Methods
CreateAlbum() : Album
CreateEvent() : Event
CreateFriendList() : FriendList
createOrPost<T>() : T
InitializeAfterSet() : void
PostLink() : Link
PostPhoto() : Post (+ 1 overload)
PostStatus() : Status
ToString() : string
User()
Nested Types
Link
Class
→ PostedItem
Properties
Description : string
IconURL : string
Message : string
Name : string
PictureURL : string
URL : string
Methods
ToString() : string
Album
Class
→ PostedItem
Fields
Properties
Count : long?
CoverPhotoID : string
Description : string
ImageAlbum : Image
ImageSmall : Image
ImageThumb : Image
Link : string
Location : string
Name : string
Photos : FacebookObjectCollection<Photo>
PictureAlbumURL : string
PictureSmallURL : string
Pictures : PicturesCollection
PictureThumbURL : string
PrivacySettings : string
Type : eType
UpdateTime : DateTime?
Methods
InitializeAfterSet() : void
ToString() : string
UploadPhoto() : Photo (+ 1 overload)
Nested Types
Checkin
Class
→ PostedItem
Fields
Properties
Message : string
Place : Place
TaggedUsers : FacebookObjectCollection<User>
Methods
ToString() : string
Status
Class
→ PostedItem
Properties
Message : string
UpdateTime : DateTime?
Methods
ToString() : string
Comment
Class
→ PostedItem
Properties
LikedByUser : bool
LikesCount : long?
Message : string
Methods
Delete() : bool
ToString() : string
Video
Class
→ PostedItem
Fields
Properties
Description : string
EmbedHtml : string
IconURL : string
Name : string
PictureURL : string
Tags : FacebookObjectCollection<TagInfo>
UpdateTime : DateTime?
URL : string
Methods
ToString() : string
Post
Class
→ PostedItem
Fields
Properties
Caption : string
Description : string
IconURL : string
LikedBy : FacebookObjectCollection<User>
LikesCount : long
Link : string
Message : string
Name : string
ObjectID : string
PictureURL : string
Source : string
TargetUsers : FacebookObjectCollection<User>
Type : eType?
UpdateTime : DateTime?
Nested Types
Photo
Class
→ PostedItem
Fields
Properties
Height : long
IconURL : string
ImageAlbum : Image
ImageNormal : Image
Images : FacebookObjectCollection<FBImage>
ImageThumb : Image
Link : string
Name : string
PictureAlbumURL : string
PictureNormalURL : string
Pictures : PicturesCollection
PictureThumbURL : string
PositionInAlbum : long?
Tags : FacebookObjectCollection<PhotoTag>
ThumbURL : string
UpdateTime : DateTime?
URL : string
Width : long
Methods
InitializeAfterSet() : void
Tag() : bool
ToString() : string
UpdateTag() : bool
PicturesCollection
Class
Fields
m_Images : Dictionary<string, Image>
m_ImageURLs : Dictionary<string, string>
m_Owner : FacebookObject
Properties
PictureUrl : string
this : Image
Methods
GetPictureUrl() : string
LoadPictureAsync() : void
LoadPicturesAsync() : void
PicturesCollection()
Event
Class
→ FacebookObject
Fields
Properties
AttendingUsers : FacebookObjectCollection<User>
DeclinedUsers : FacebookObjectCollection<User>
Description : string
EndTime : DateTime?
ImageLarge : Image
ImageNormal : Image
ImageSmall : Image
ImageSquare : Image
InvitedUsers : FacebookObjectCollection<User>
LinkToFacebook : string
Location : string
MaybeAttendingUsers : FacebookObjectCollection<User>
Name : string
NotYetRepliedUsers : FacebookObjectCollection<User>
Owner : User
PictureLargeURL : string
PictureNormalURL : string
Pictures : PicturesCollection
PictureSmallURL : string
PictureSquareURL : string
Privacy : ePrivacy
StartTime : DateTime?
TimeString : string
UpdateTime : DateTime?
UpdateTimeString : string
WallPosts : FacebookObjectCollection<Post>
Methods
InitializeAfterSet() : void
PostLink() : string
PostMessage() : string
Respond() : bool
ToString() : string
Nested Types
ePrivacy
Enum
eRsvpType
Enum


DynamicWrapper
Abstract Class
Fields
m_DynamicData : dynamic
s_FieldsToLoadFull : Dictionary<eLoadOptions, string>
Properties
FieldsToLoad : Dictionary<eLoadOptions, string>
LoadState : eLoadOptions
Methods
GetWrapped<T>() : T
InitializeAfterSet() : void
Set() : void
WrapOrGet<T>() : T
Nested Types
eLoadOptions
Enum
FbWrapper
Class
Methods
getLoggedInUser() : User
Login() : LoginResult (+ 1 overload)
LoginResult
LoginResult
Class
Properties
AccessToken : string
ErrorMessage : string
Place
Class
DynamicWrapper
Fields
Properties
Location : Location
Name : string
Methods
ToString() : string
Location
Class
DynamicWrapper
Properties
City : string
Country : string
Latitude : double
Longitude : double
Methods
ToString() : string
Poke
Class
DynamicWrapper
Fields
Properties
CreatedTime : DateTime?
From : User
To : User
Methods
ToString() : string
FBImage
Class
DynamicWrapper
Properties
Height : int
Source : string
Width : int
TagInfo
Class
DynamicWrapper
Fields
Properties
CreatedTime : DateTime?
User : User
FacebookObject
Abstract Class
DynamicWrapper
Properties
Id : string
Methods
CreateOrGetCollection<T>() : FacebookObjectCollection<T>
DeleteObject() : bool
Fetch() : void (+ 1 overload)
RetrieveCollection<T>() : FacebookObjectCollection<T>
PhotoTag
Class
TagInfo
Properties
XCoord : float
YCoord : float
InboxThread
Class
FacebookObject
Fields
Properties
From : User
Message : string
MessageCount : long?
Snippet : string
Subject : string
UnreadCount : long?
UpdatedTime : DateTime?
Methods
ToString() : string
Message
Class
FacebookObject
Fields
Properties
CreatedTime : DateTime?
From : User
Text : string
Methods
ToString() : string
PostedItem
Abstract Class
FacebookObject
Fields
Properties
CreatedTime : DateTime?
FieldsToLoad : Dictionary<eLoadOptions, string>
From : User
Methods
Comment() : Comment
Like() : bool
PostedItem()
Unlike() : bool
FriendList
Class
FacebookObject
Fields
Properties
Name : string
Methods
AddMemeber() : bool
Delete() : bool
RemoveMemeber() : bool
ToString() : string
User
Class
FacebookObject
Fields
Properties
Bio : string
Birthday : string
Email : string
FieldsToLoad : Dictionary<eLoadOptions, string>
FirstName : string
Friends : FacebookObjectCollection<User>
Gender : eGender?
ImageLarge : Image
ImageNormal : Image
ImageSmall : Image
ImageSqaure : Image
InboxThreads : FacebookObjectCollection<InboxThread>
InterestedIn : eGender[]
LastName : string
Link : string
Locale : string
MessageUpdates : FacebookObjectCollection<Message>
MiddleName : string
Name : string
PhotosTaggedIn : FacebookObjectCollection<Photo>
PictureLargeURL : string
PictureNormalURL : string
PictureSmallURL : string
PictureSqaureURL : string
Quotes : string
RelationshipStatus : eRelationshipStatus?
Religion : string
SignificantOther : User
TimeZone : long?
UpdateTime : DateTime?
UserName : string
Verfied : bool?
Methods
CreateAlbum() : Album
CreateEvent() : Event
CreateFriendList() : FriendList
createOrPost<T>() : T
InitializeAfterSet() : void
PostLink() : Link
PostPhoto() : Post (+ 1 overload)
PostStatus() : Status
ToString() : string
User()
Nested Types
Members :...
Pokes...
FriendLists...
LikedBy...
Comments :...
PostedLinks :...
Messages :...
Link
Class
PostedItem
Properties
Description : string
IconURL : string
Message : string
Name : string
PictureURL : string
URL : string
Methods
ToString() : string
Album
Class
PostedItem
Fields
Properties
Count : long?
CoverPhotoID : string
Description : string
ImageAlbum : Image
ImageSmall : Image
ImageThumb : Image
Link : string
Location : string
Name : string
PictureAlbumURL : string
Pictures : PicturesColleciton
PictureSmallURL : string
PictureThumbURL : string
PrivacySettings : string
Type : eType
UpdateTime : DateTime?
Methods
InitializeAfterSet() : void
ToString() : string
UploadPhoto() : Photo (+ 1 overload)
Nested Types
Checkin
Class
PostedItem
Fields
Properties
Message : string
Place : Place
Methods
ToString() : string
Status
Class
PostedItem
Properties
Message : string
UpdateTime : DateTime?
Methods
ToString() : string
Comment
Class
PostedItem
Properties
LikedByUser : bool
LikesCount : long?
Message : string
Methods
Delete() : bool
ToString() : string
Statuses :...
Checkins...
TaggedUsers :...
Pictures
PicturesColleciton
Class
Fields
m_Images : Dictionary<string, Image>
m_ImageURLs : Dictionary<string, string>
m_Owner : FacebookObject
Properties
PictureUrl : string
this : Image
Methods
GetPictureUrl() : string
LoadPictureAsync() : void
LoadPicturesAsync() : void
PicturesColleciton()
Post
Class
PostedItem
Fields
Properties
Caption : string
Description : string
IconURL : string
LikesCount : long
Link : string
Message : string
Name : string
ObjectID : string
PictureURL : string
Source : string
Type : eType?
UpdateTime : DateTime?
Nested Types
LikedBy...
TargetUsers :...
NewsFeed...
Posts :...
PostsTaggedIn :...
WallPosts :...
Albums :...
Events...
Video
Class
PostedItem
Fields
Properties
Description : string
EmbedHtml : string
IconURL : string
Name : string
PictureURL : string
Tags : FacebookObjectCollection<TagInfo>
UpdateTime : DateTime?
URL : string
Methods
ToString() : string
Photos :...
Photo
Class
PostedItem
Fields
Properties
Height : long
IconURL : string
ImageAlbum : Image
ImageNormal : Image
Images : FacebookObjectCollection<FBImage>
ImageThumb : Image
Link : string
Name : string
PictureAlbumURL : string
PictureNormalURL : string
Pictures : PicturesColleciton
PictureThumbURL : string
PositionInAlbum : long?
Tags : FacebookObjectCollection<PhotoTag>
ThumbURL : string
UpdateTime : DateTime?
URL : string
Width : long
Methods
InitializeAfterSet() : void
Tag() : bool
ToString() : string
UpdateTag() : bool
Videos...
Event
Class
FacebookObject
Fields
Properties
Description : string
EndTime : DateTime?
ImageLarge : Image
ImageNormal : Image
ImageSmall : Image
ImageSqaure : Image
LinkToFacebook : string
Location : string
Name : string
PictureLargeURL : string
PictureNormalURL : string
PictureSmallURL : string
PictureSqaureURL : string
Privacy : ePrivacy
StartTime : DateTime?
TimeString : string
UpdateTime : DateTime?
UpdateTimeString : string
Methods
InitializeAfterSet() : void
PostLink() : string
PostMessage() : string
Respond() : bool
ToString() : string
Nested Types
ePrivacy
Enum
eRsvpType
Enum
Owner
DeclinedUsers :...
AttendingUsers...
InvitedUsers :...
MaybeAttendingUsers :...
NotYetRepliedUsers :...
WallPosts :...