

### Creating a facebook desktop application using Facebook SDK and FbGraphApiWrapper (.NET Framework 4, WinForms, Visual Studio 2010)

#### 1. Creating an application-account in Facebook

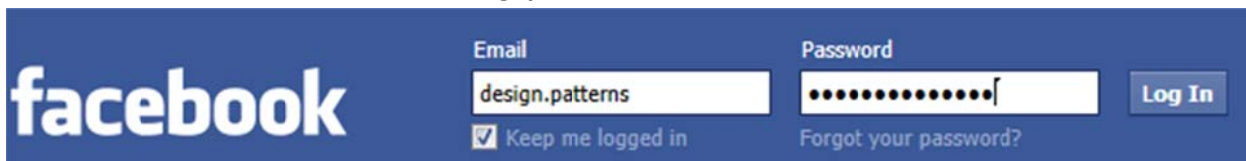
In order to create an application that communicates with Facebook on behalf of your user, you must first create an application-account in Facebook:

- a. You must have a Facebook account in order to create an application account which will be created under your Facebook account.

You should use the Course's Facebook account-

Account's **UserName**: **design.patterns**

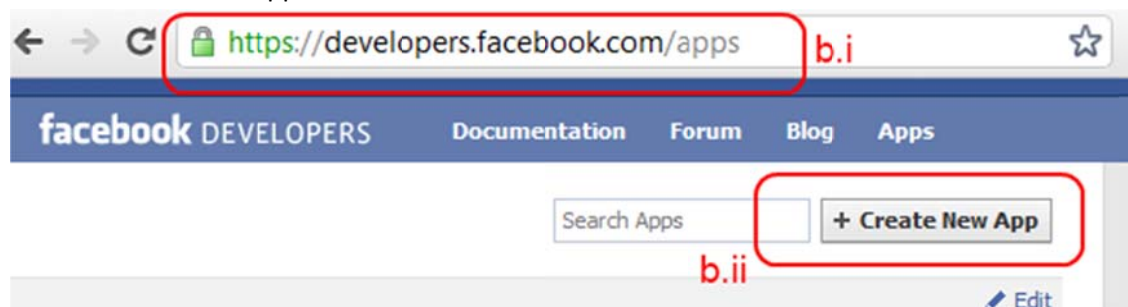
Account's **Password**: **designpatterns**



- b. Create an Application Account:

- i. Go to <https://developers.facebook.com/apps>

- ii. Create a new Application Account

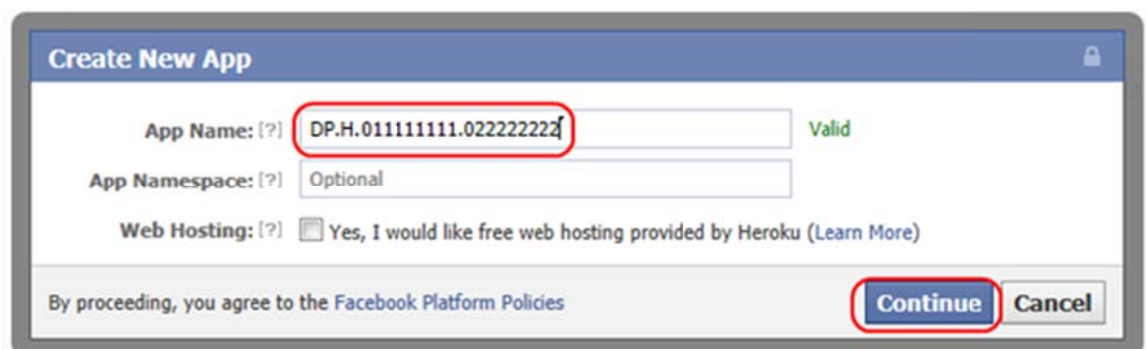


- iii. Give the application a name using the following format:

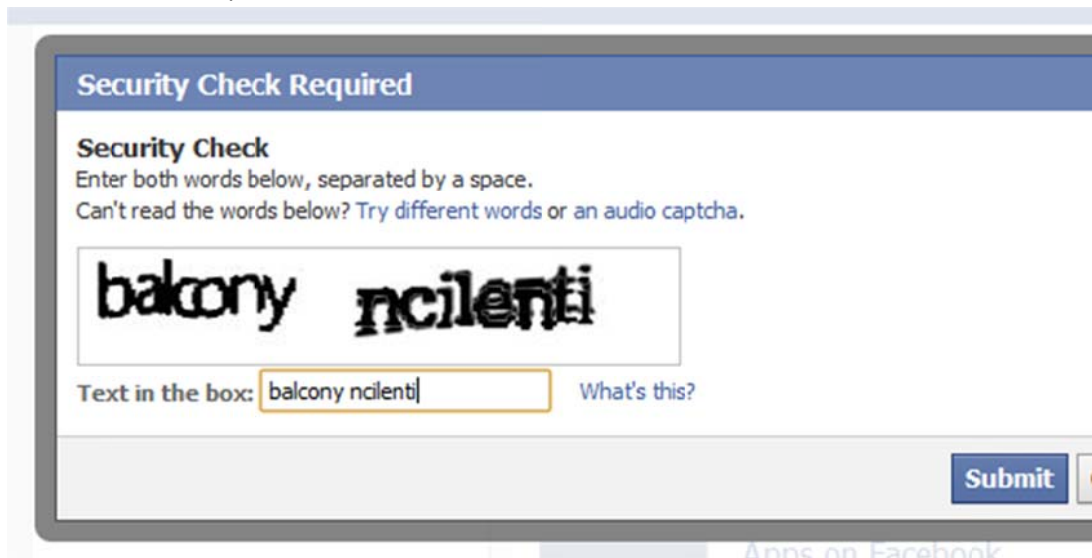
**DP.H.B12.01111111.02222222**

01111111 is the first student's ID number (9 digits!)

02222222 is the second student's ID number (9 digits!)



iv. Enter the captcha:



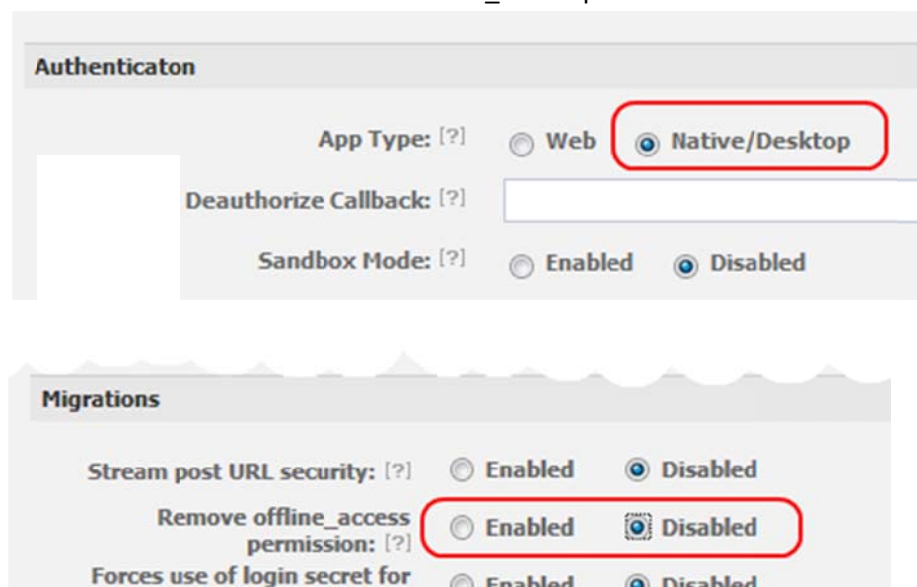
A security check form titled "Security Check Required". It asks the user to enter two words from an image: "balcony" and "ncilenti". The words are displayed in a stylized, slightly distorted font. Below the image, there is a text input field containing "balcony ncilenti". To the right of the input field is a link that says "What's this?". At the bottom right of the form is a blue "Submit" button.

v. Save the App ID for later and then go to 'Advanced':



The Facebook App Settings page for an app named "DP.H.01111111.02222222". The left sidebar shows the "Settings" menu with "Basic", "Auth Dialog", "Advanced", "Open Graph", and "Roles". The "Advanced" option is highlighted with a red circle. The main content area shows the app's icon, name, and ID. The "App ID" is "136619623131579" and is circled in red. The "App Secret" is "534588ff457b77ab2cc3036ac253774f" and is also circled in red. There is a "(reset)" link next to the app secret and an "(edit icon)" link next to the app icon.

vi. In the 'Advanced' section, **select the Native/Desktop** application type and **select 'Disabled'** in "Remove offline\_access permission":



The "Advanced" section of the Facebook App Settings page. It contains two main sections: "Authentication" and "Migrations". In the "Authentication" section, the "App Type" is set to "Native/Desktop" (circled in red) instead of "Web". The "Deauthorize Callback" field is empty. The "Sandbox Mode" is set to "Disabled" (circled in red) instead of "Enabled". In the "Migrations" section, the "Remove offline\_access permission" is set to "Disabled" (circled in red) instead of "Enabled". Other options include "Stream post URL security" (Disabled), "Forces use of login secret for" (Disabled), and "Enabled" options for "Stream post URL security", "Remove offline\_access permission", and "Forces use of login secret for".

and the hit "Save Changes".

## 2. Creating a .NET 4 WinForms application using the FbGraphApiWrapper.dll assembly

- a. In Visual Studio 2010, create a new **.NET 4** WinForms project
- b. 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).
- c. Use the static login method  
`LoginResult result = FacebookWrapper.FBService.Login("192940970765612",`  
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](#).
- d. 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.
- e. 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.

## 3. 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)

