

## 1 Lab session – Facebook and Android

This lab session is intended to show how to create mobile social applications for the Android platform, based on Facebook. It is composed of two steps:

- **Part 1:** In the first part of this lab session you are going to create, step by step and using a provided code-templates, an **Application interacting with Facebook**. You will learn which are basic mandatory steps to be performed in order to use **Facebook API** inside your Android application.

*Time approximately requested for completing this part: <2 hours*

- **Part 2 (assignment):** *[Requires the Exercise 1 to be complete and working]* In the second part of the lab session you are going to extend the application built previously, adding some features which have to be coded from scratch (no templates of solution provided for this part).

*Time approximately requested for completing this part: <1 hour*

- **Part 3 : extend your mobserv challenge project to support facebook (bonous)**

For developing and testing this lab session you have to work in groups of 2-3 people. Your group will be using a single test facebook user profile, among the ones provided hereafter. Those users are special sandboxed ones, associated to the App we are developing.

*One import remark is that, even though the App will be built on several platforms (the computer for each group working on), however it is a single App, that has been previously created and described on <https://developers.facebook.com/>.*





Our common Facebook App has the following properties (see also <https://developers.facebook.com/docs/getting-started/facebook-sdk-for-android/3.0/>):

- **Display name:** myfbapp  
The displayed name of the application.
- **Namespace:** mobserv  
A unique string identifying the application, used by some Facebook APIs you maybe would like to experiment.
- **App ID:** 109994289166606  
The unique ID identifying the application, shared by all the groups (as we have a single App for everybody). It is not secret, and in general can be disclosed.
- **SecretKey:** d23d16f61bca3f4d03adcc28e1dfffb36  
This is a randomly generated key which is secret, i.e. it cannot be disclosed to anyone else (used for identifying the client calling the Facebook APIs).
- A list of **hash keys** with which the “origin of the application” (i.e. the platform on which the apk - Android Package - has been generated) can be verified. For each platform (i.e. the pc you are using) and user id. there’s a unique hash key: it is the

hash of the debug key in your own Android SDK, which is used by Eclipse to sign the apk you build. We will see later how to get it.

This means that only platforms whose hash key has been added to this list will be able to generate valid apks for this application. This feature acts as a “Proof of origin” for the apk, and it can be used instead of the Secret Key.

### 1.1 Get your hash key

A basic prerequisite for getting the hash key is to have OpenSSL installed on your machine. The actual installation process depends on your OS: download its last version from [www.openssl.org](http://www.openssl.org) and check the README and INSTALL.X files for instructions about how to install it. For all the linux machine, openssl has been already installed.

Enter the shell and execute the following command (the path of the keystore actually depends on your own platform):

**Linux/MacOSX:**

```
$ keytool -exportcert -alias androiddebugkey -keystore ~/.android/debug.keystore  
| openssl sha1 -binary | openssl base64
```

**Windows:**

```
$ keytool -exportcert -alias androiddebugkey -keystore  
%HOMEPATH%\android\debug.keystore | openssl sha1 -binary | openssl base64
```

It gets the debug key from the Android SDK of your platform and you will be asked for the password: the default password is “**android**”. Then it calculates the hash of such a key, using SHA1 Algorithm and finally it prints out the result, encoded in Base64 (to make it readable). This is the hash key you need to set in your app parameters in facebook (see figure above).

Please send the hash key by mail to [navid.nikaein@eurecom.fr](mailto:navid.nikaein@eurecom.fr), so that it will be added to the list.

### 1.2 Part 1: create a mobile social application

To create an Android App using Facebook, you have to deal with Facebook SDK, in order to be able to perform the calls to the Facebook API needed by your App:

1. To perform the **authentication**, :
  - a. **normal login**: your App is logging in by itself, asking to the user the username and the password (in any case through an Android Activity dynamically generated by the Facebook SDK);
  - b. or **Single Sign On (SSO)**: your App is checking if at least another application on the smartphone has already performed the login during the current session. In this case, if the application has been previously authorized to have access to the user facebook data (friend list, posts, etc...), the login will be performed automatically, without any login screen.  
*In this lab session we will be using the SSO authentication.*
2. [after authentication] For performing the **call to Facebook APIs** of interest, like getting the list of friends: [graph.facebook.com/me/friends](https://graph.facebook.com/me/friends).

### 1.2.1 Step 1.0: configure the environment

- Create an avd for android 4.1.2, and start it
- Install facebook app on the emulator
  - `./adb install packages/android-sdk/facebook-android-sdk-3.0.2.b/bin/FBAndroid-1.9.10.apk`

Note: The last version of the facebook SDK (3.0.2.b) can be found at `/packages/android-sdk/facebook-sdk-android`. Otherwise, you can download facebook android sdk from <https://developers.facebook.com/android/>

These steps are also explained in step 1 and 2 here:

<https://developers.facebook.com/docs/getting-started/facebook-sdk-for-android/3.0/>

### 1.2.2 Step 1.1: get the Hackbook example working

Perform following tasks of the **steps 3 and 4** at [1]:

- Import the project in your workspace the projects `FacebookSDK` and `Hackbook` from the `facebook-android-sdk-<version>` folder. You can also import other projects (like `HelloFacebookSample`) to check some simple examples of fully working Apps interacting with Facebook;

- Check the Eclipse Console: if you get a message like the one below, fix the issue as suggested:

```
[XXXX-XX-XX XX:XX:XX - FacebookSDK] Android requires compiler compliance level 5.0 or 6.0. Found 'X' instead. Please use Android Tools > Fix Project Properties.
```

Hint: if you get some 'Must Override a Superclass Method' errors after importing such projects into Eclipse, please check the following link to see how to change your Java Compiler Compliance Level from 1.5 to 1.6+:

<http://stackoverflow.com/questions/1678122/must-override-a-superclass-method-errors-after-importing-a-project-into-eclips>

- Run the app on the Android Emulator (target: Android 4.x) and check it's working properly. Try to login using the mail address of your group and "eurecom" as password.

**Utilisateurs de Test**
Créer · Voir tous les rôles

Select All
Select None

Enable Ticker
Make Friends
Add to Other Apps
Supprimer

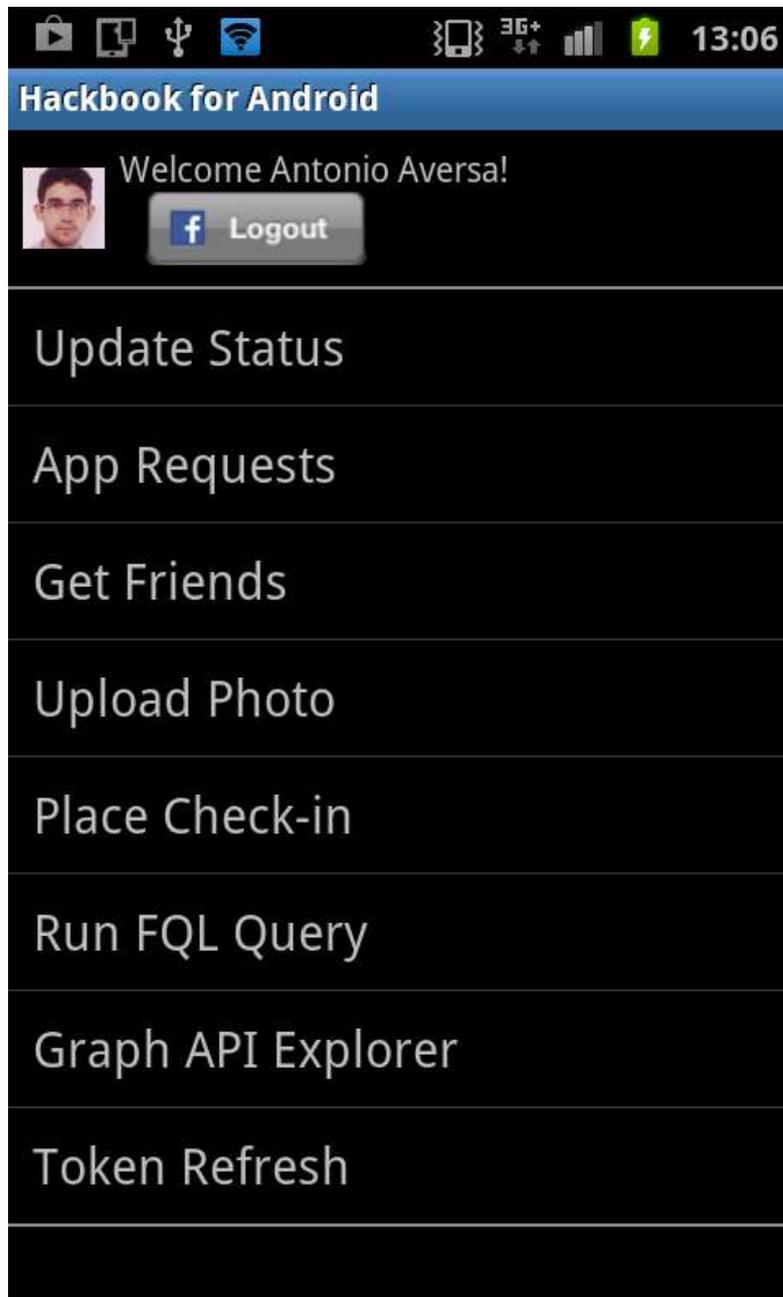
	Name	ID	Email	
<input type="checkbox"/>	mobserv group j	100004725640460	mobserv_mbvhlhvz_j@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group i	100004714660677	mobserv_ybqqcj_i@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group h	100004744900308	mobserv_wceondd_h@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group g	100004702241133	mobserv_mxbjnwq_g@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group f	100004759924874	mobserv_omawwxj_f@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group e	100004706020837	mobserv_nztxepo_e@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group d	100004761484887	mobserv_hnwodox_d@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group c	100004725100689	mobserv_ypfjst_c@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group b	100004716187490	mobserv_ngzgmxt_b@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv test	100004713637536	mobserv_jyhwro_test@tfnw.net	<a href="#">Afficher plus</a>
<input type="checkbox"/>	mobserv group a	100004766131757	mobserv_ibddipt_a@tfnw.net	<a href="#">Afficher plus</a>

You will end up with the following error:



This means you need to set the right APP\_ID in the code and have your hash key added in the list....

- Get the Hackbook project fully working.



*Hint: if you have some issues with SSO, try to check the following link:*  
<http://sean.lyn.ch/2011/07/android-the-facebook-sdk-sso-and-you/>

### 1.2.3 Step 1.2: adapt Hackbook to your needs

Hackbook is a good starting point for checking a lot of Facebook APIs, but we want to get a very simple App which just allow the user to post a message on the wall of a friend of his. To achieve this result we want:

- **Step 1:** To remove the splash screen shown at the App startup: the main menu of the Hackbook activity has to be shown instead. Check the manifest.xml file and be sure to have removed the corresponding Java class and the layout xml file;
- **Step 2 :** To remove the list of menu items shown in the main menu, replacing it with a simple button “Post a message”:

```
<Button
    android:id="@+id/buttonPostAMessage"
    android:layout_width="306dp"
    android:layout_height="wrap_content"
    android:layout_weight="0.01"
    android:text="Post A Message"
    android:onClick="doPostAMessage"/>
```

Remove the field `private ListView list` in `Hackbook.java` and the related assignments.

Define the method `public void doPostAMessage(View view)` in `Hackbook.java`, performing the action previously performed by the “Get Friends” list item.

*Hint: check layout-land & layout-port directories while looking for the layout for the Hackbook activity*

**Questions:**

- why there are two main.xml files?
  - how to customize the background of the App depending on the country of the app?
  - how many different values for the property orientation exist?
  - what would be the behavior of the App if the orientation were “square”?
- **Step 3:** when the user click on the new button “Post A Message”, a small window “Graph API or FQL Query” is shown:
    - identify the code which is generating such a window;
    - remove such window and execute the request “get friends” using directly a FQL Query;

*Hint: don't forget to remove `dialog.dismiss()`; in the code of the class `FriendsRequestListener`*

- what does FQL mean? What is it useful for?
- which is the FQL query performed for getting the friend list?
- Navigate the code (using F3), class by class, starting from the request to the `asyncRunner` object to execute such a query:
  - which is the method that actually calls the Facebook API?
  - what kind of protocol does it use? Show the first line of code which start the deliver of the request.

### 1.3 Part 2 (assignment): extend your App using other Facebook APIs

In this exercise we want to implement a new feature in the Hackbook App: we want to use the facebook API to retrieve the email address for each Facebook friend, and, for each of them, add a contact in the contact list. The step you have to go through are:

- add a second button in the Hackbook main activity, with a new onClick handler;
- in this onClick handler:
  - erase all the contact in the Contact List [2];
  - use the right Facebook API, in order to retrieve the friend list (with emails) [3];
  - iterate over the result and add a new contact for each of them [4].

### 1.4 Part 3 extend your mobserv challenge to support face book

This is optional feature you may add to your mobserv challenge project, which brings you %5 bonous.

[2] <http://stackoverflow.com/questions/527216/how-to-remove-a-contact-programmatically-in-android>

[3] <http://shikii.net/blog/getting-a-list-of-a-facebook-users-friends-with-their-email-addresses/>

[4] <http://stackoverflow.com/questions/4744187/how-to-add-new-contacts-in-android>