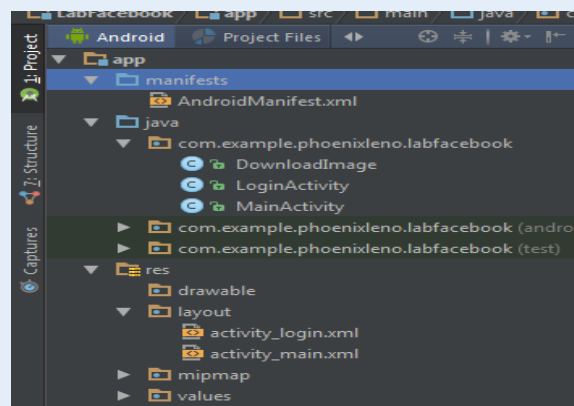


Requisitos:

1. Emulador o dispositivo con API de nivel 15 o más.
2. Aplicación oficial de Facebook Instalada

Agregar el SDK de Facebook

1. Cree un nuevo proyecto en Android Studio, con un API nivel 15 o superior (ice cream sandwich mínimo) y un empty activity que tenga el nombre de “LoginActivity”.
2. Agregue un empty activity llamado “MainActivity” y una clase llamada “DownloadImage”.



3. Una vez creado el proyecto, agregue el siguiente código en el build.gradle de la aplicación:

```
repositories {  
    mavenCentral()  
}  
  
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
    testCompile 'junit:junit:4.12'  
    compile 'com.android.support:appcompat-v7:23.3.0'  
    compile 'com.android.support:design:23.+'  
    compile 'com.facebook.android:facebook-android-sdk:4.+'  
}
```

4. Agregue e inicialice el SDK en el onCreate de la aplicación

```
package com.example.phoenixleno.labfacebook;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import com.facebook.FacebookSdk;

public class MainActivity extends AppCompatActivity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        FacebookSdk.sdkInitialize(getApplicationContext());
    }
}
```

5. Copie el siguiente código en el styles.xml

```
<resources>

    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="windowActionBar">false</item>
        <item name="windowNoTitle">true</item>
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>
    <style name="AppTheme.NoActionBar">
        <item name="windowActionBar">false</item>
        <item name="windowNoTitle">true</item>
    </style>
    <style name="AppTheme.AppBarOverlay" parent="ThemeOverlay.AppCompat.Dark.ActionBar" />
    <style name="AppTheme.PopupOverlay" parent="ThemeOverlay.AppCompat.Light" />

</resources>
```

Agregar Facebook App ID

1. Agregue en el archivo strings.xml las siguientes líneas, en este espacio se especificará el id de Facebook, para asociar el proyecto Android con la aplicación de Facebook, que se creará más adelante:

```
<resources>
    <string name="app_name">Facebook</string>
    <string name="action_settings">LogOut</string>
    <string name="title_activity_main">MainActivity</string>
    <string name="app_id"></string>
</resources>
```

2. En el archivo AndroidManifest.xml agregue el permiso para acceder a Internet

```
<uses-permission android:name="android.permission.INTERNET" />
```

3. Ahora agregue el elemento que recibirá el id de la aplicación, junto con el activity de Facebook y el provider para adjuntar imágenes.

```
android:theme="@style/AppTheme">
<activity android:name=".LoginActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>

<activity
    android:name=".MainActivity"
    android:label="@string/app_name"
    android:theme="@style/AppTheme" >
</activity>

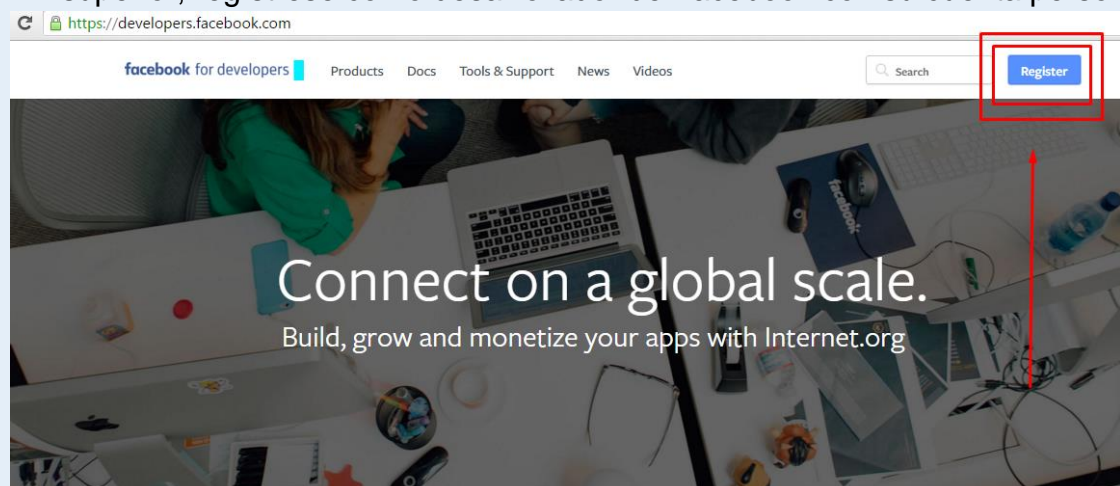
<meta-data
    android:name="com.facebook.sdk.ApplicationId"
    android:value="@string/app_id" />

<activity
    android:name="com.facebook.FacebookActivity"
    android:label="@string/app_name"
    android:screenOrientation="portrait"/>

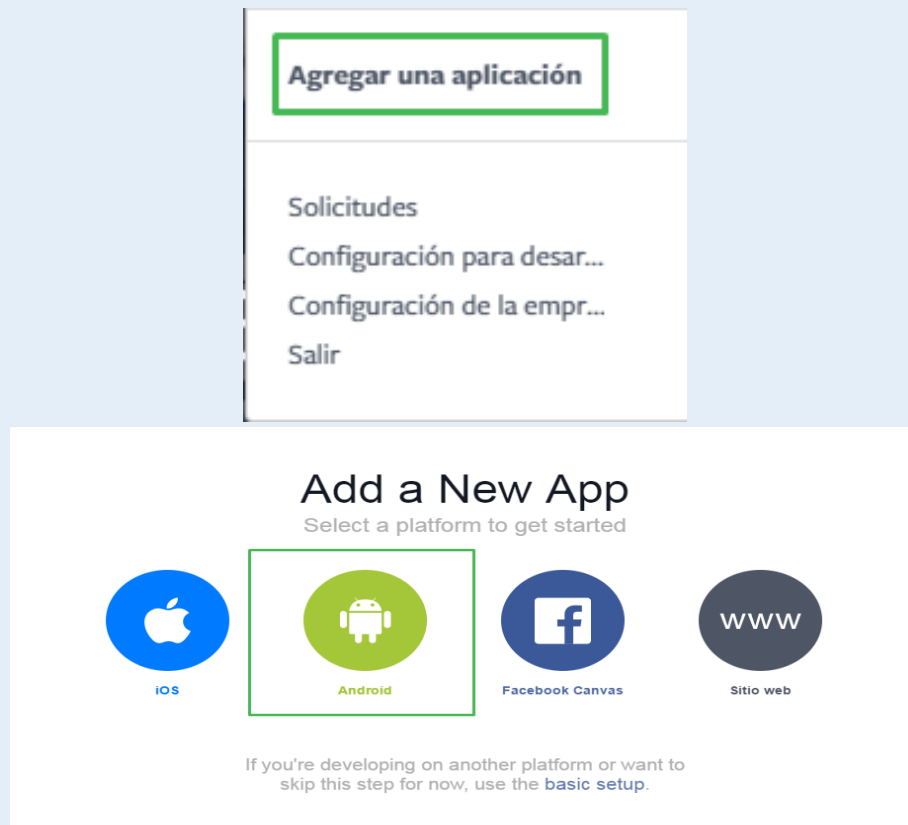
<provider
    android:authorities="com.facebook.app.FacebookContentProvider"
    android:name="com.facebook.FacebookContentProvider"
    android:exported="true"/>
</application>
</manifest>
```

Registrar la aplicación Facebook

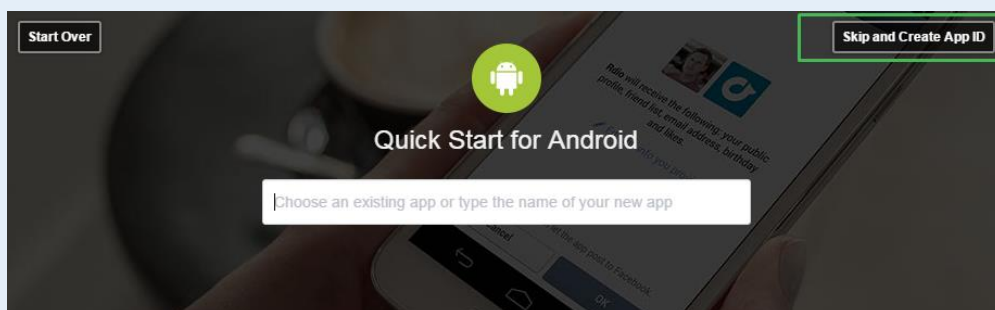
1. Vaya al siguiente link <https://developers.facebook.com/> y en la parte derecha superior, regístrese como desarrollador de Facebook con su cuenta personal.



2. Después de registrarse, se iniciará sesión con su perfil, haga clic en “Mis Aplicaciones” y seleccione “Agregar una aplicación”, luego elija Android.



3. En la página que le muestra seleccione “Skip and Create App ID”, luego complete el formulario que se despliega. El nombre puede o no ser diferente al de su proyecto.



Crear un nuevo identificador de la aplicación

Empezar a integrar Facebook en tu aplicación o sitio web

Nombre para mostrar

Correo electrónico de contacto

Espacio de nombres

☐ No ¿Es esta una versión de prueba de otra aplicación? [Más información.](#)


Categoría

Deportes ▼

Al continuar, aceptas las [Políticas de la plataforma de Facebook](#)

4. Después de esto se le mostrará una página con el ID de su aplicación, cópielo y péguelo en el archivo strings.xml del proyecto, en el string “app_id” creado anteriormente

Panel



TutorialLogin ○

Esta aplicación está en modo de desarrollo y solo la pueden usar los administradores, desarrolladores y evaluadores [?]

Versión de la API [?]

v2.6

Identificador de la aplicación

144197909313878

Clave secreta de la aplicación

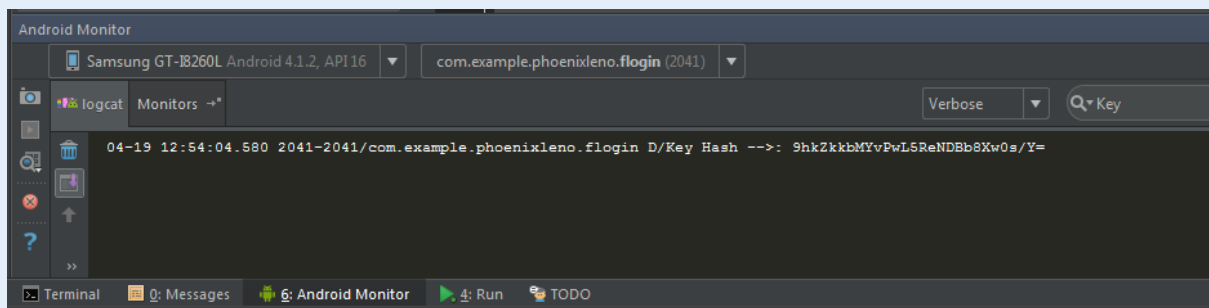
.....

```
<resources>
  <string name="app_name">Facebook</string>
  <string name="action_settings">LogOut</string>
  <string name="title_activity_main">MainActivity</string>
  <string name="app_id">144197909313878</string>
</resources>
```

Generar Key Hash

1. Como medida de seguridad se utiliza una clave generada a partir de un hash para que la aplicación se pueda conectar con Facebook, esta llave se puede generar ejecutando la siguiente función en el proyecto de Android en el onCreate y revisando el resultado en el log:

```
public void printKeyHash() {  
    try {  
        PACKAGE_NAME = getApplicationContext().getPackageName();  
        PackageInfo info = getPackageManager().getPackageInfo(PACKAGE_NAME, PackageManager.GET_SIGNATURES);  
        for (Signature signature : info.signatures) {  
            MessageDigest md = MessageDigest.getInstance("SHA");  
            md.update(signature.toByteArray());  
            Log.d("Key Hash -->", Base64.encodeToString(md.digest(), Base64.DEFAULT));  
        }  
    } catch (PackageManager.NameNotFoundException e) {  
    } catch (NoSuchAlgorithmException e) {  
    }  
}
```

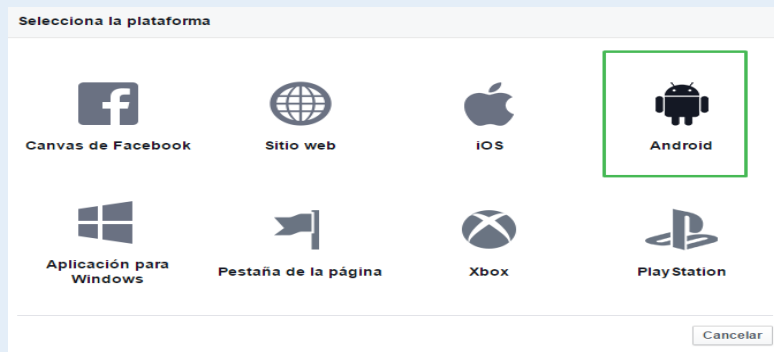


2. Ahora, en la página de la aplicación en Facebook Developer, vaya a “Configuración”, luego haga clic en “Agregar plataforma” y seleccione Android.

The screenshot shows the Facebook Developer application configuration page for 'TutorialLogin'. The 'Básica' (Basic) tab is selected. The page contains the following fields:

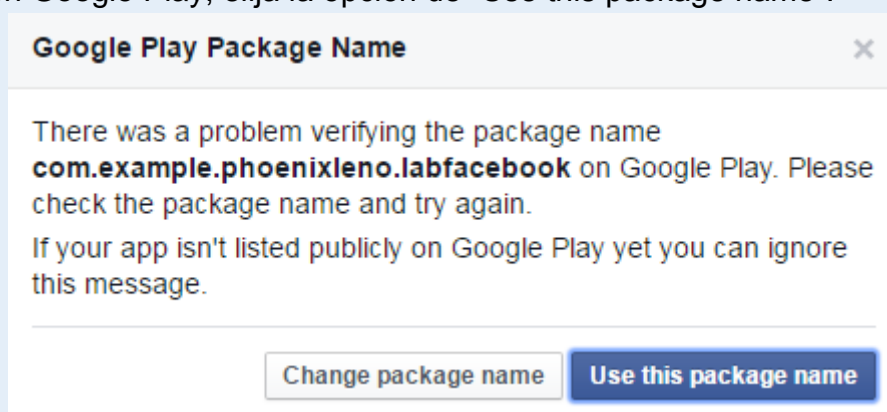
- Identificador de la aplicación: 144197909313878
- Clave secreta de la aplicación: [Redacted] (with a 'Mostrar' button)
- Nombre para mostrar: TutorialLogin
- Espacio de nombres: [Empty field]
- Dominios de la aplicación: [Empty field]
- Correo electrónico de contacto: correo@gmail.com

At the bottom, there is a button labeled '+ Agregar plataforma' which is highlighted with a green box.



3. A continuación, copie los 28 caracteres del hash key generado en el proyecto y péguelo en el espacio correspondiente en la página. Además, llene los nombres de las rutas de su proyecto y su activity, marque “Sí” en Inicio de sesión único y, por último, guarde los cambios.

Le saldrá un mensaje de la página debido a que el nombre del paquete no se encuentra en Google Play, elija la opción de “Use this package name”.



4. Finalmente en la sección de “Revisión de aplicaciones”, haga la aplicación pública marcando “Sí”, de otro modo no se podrá iniciar sesión en la aplicación con otra cuenta

The screenshot shows the Facebook Developer Portal interface for an application named 'TutorialLogin'. On the left is a sidebar menu with options: 'Panel', 'Configuración', 'Revisión de aplicaciones' (highlighted with a blue star), 'Detalles de la aplicación', and 'Roles'. The main content area has a header question: '¿Quieres que TutorialLogin sea una aplicación pública?'. Below this is a toggle switch set to 'Sí' and a status message: 'Tu aplicación está actualmente **activa** y disponible para el público.' Below that is a section titled 'Enviar elementos para aprobación' with a sub-note about Facebook integrations and a button labeled 'Iniciar una solicitud'.

Inicio de sesión con Facebook

1. Modifique el archivo xml correspondiente al LoginActivity para que quede de la siguiente manera:

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent" android:fitsSystemWindows="true"
    tools:context=".LoginActivity">

    <android.support.design.widget.AppBarLayout android:layout_height="wrap_content"
        android:layout_width="match_parent" android:theme="@style/AppTheme.AppBarOverlay">

        <android.support.v7.widget.Toolbar android:id="@+id/toolbar"
            android:layout_width="match_parent" android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary" app:popupTheme="@style/AppTheme.PopupOverlay" />

    </android.support.design.widget.AppBarLayout>
    <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">
```

```

</android.support.design.widget.AppBarLayout>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <com.facebook.login.widget.LoginButton
        android:id="@+id/login_button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:gravity="center"
        android:layout_margin="4dp"
        android:paddingTop="12dp"
        android:paddingBottom="12dp"/>

</LinearLayout>
</android.support.design.widget.CoordinatorLayout>

```

2. Ahora modifique el archivo xml del activity MainActivity para que quede así:

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent" android:fitsSystemWindows="true"
    tools:context=".MainActivity">

    <android.support.design.widget.AppBarLayout android:layout_height="wrap_content"
        android:layout_width="match_parent" android:theme="@style/AppTheme.AppBarOverlay">

        <android.support.v7.widget.Toolbar android:id="@+id/toolbar"
            android:layout_width="match_parent" android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary" app:popupTheme="@style/AppTheme.PopupOverlay" />

    </android.support.design.widget.AppBarLayout>

    <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        xmlns:app="http://schemas.android.com/apk/res-auto" android:layout_width="match_parent"
        android:layout_height="match_parent" android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        android:paddingBottom="@dimen/activity_vertical_margin"
        app:layout_behavior="@string/appbar_scrolling_view_behavior"
        tools:showIn="@layout/activity_main" tools:context="com.sitepoint.facebook.MainActivity">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical">

```

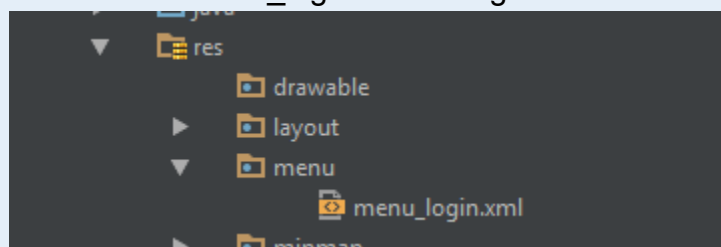
```

<TextView
    android:text="HOLA"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:textSize="20dp"
    android:layout_gravity="center_horizontal"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/nameAndSurname"
    android:textSize="22dp"
    android:textStyle="bold"
    android:layout_marginTop="10dp"
    android:layout_gravity="center_horizontal"/>
<ImageView
    android:layout_width="200dp"
    android:layout_height="200dp"
    android:id="@+id/profileImage"
    android:layout_marginTop="10dp"
    android:layout_gravity="center_horizontal"/>
</LinearLayout>
</RelativeLayout>

<android.support.design.widget.FloatingActionButton android:id="@+id/fab"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:layout_gravity="bottom|end"
    android:src="@android:drawable/ic_menu_edit" />
</android.support.design.widget.CoordinatorLayout>

```

3. En la carpeta de res, cree un directorio con el nombre “menu” y adentro de éste cree un archivo xml llamado “menu_login” con el siguiente contenido:



```

<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context="com.sitepoint.facebook.MainActivity">

    <item android:id="@+id/action_settings"
        android:title="@string/action_settings"
        android:orderInCategory="100"
        app:showAsAction="never"/>
</menu>

```

4. Copie el siguiente código en la clase DownloadImage:

```
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.AsyncTask;
import android.util.Log;
import android.widget.ImageView;

import java.io.InputStream;

public class DownloadImage extends AsyncTask<String, Void, Bitmap> {
    ImageView bmImage;

    public DownloadImage(ImageView bmImage) { this.bmImage = bmImage; }

    protected Bitmap doInBackground(String... urls) {
        String urldisplay = urls[0];
        Bitmap mIcon11 = null;
        try {
            InputStream in = new java.net.URL(urldisplay).openStream();
            mIcon11 = BitmapFactory.decodeStream(in);
        } catch (Exception e) {
            Log.e("Error", e.getMessage());
            e.printStackTrace();
        }
        return mIcon11;
    }

    protected void onPostExecute(Bitmap result) { bmImage.setImageBitmap(result); }
```

5. Dentro del LoginActivity agregue el siguiente código, y haga los imports necesarios.

```
public class LoginActivity extends AppCompatActivity {

    private CallbackManager callbackManager;
    private AccessTokenTracker accessTokenTracker;
    private ProfileTracker profileTracker;
    public static String PACKAGE_NAME;
    //Boton de login
    private FacebookCallback<LoginResult> callback = new FacebookCallback<LoginResult>() {
        @Override
        public void onSuccess(LoginResult loginResult) {
            Profile profile = Profile.getCurrentProfile();
            nextActivity(profile);
        }
        @Override
        public void onCancel() { }
        @Override
        public void onError(FacebookException e) { }
    };

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        FacebookSdk.sdkInitialize(getApplicationContext()); //Inicializa el SDK de facebook

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        printKeyHash(); //Metodo para imprimir el hash
        callbackManager = CallbackManager.Factory.create();
        accessTokenTracker = new AccessTokenTracker() {
            @Override
            protected void onCurrentAccessTokenChanged(AccessToken oldToken, AccessToken newToken) {
            }
        }
    }
}
```

```

};
profileTracker = new ProfileTracker() {
    @Override
    protected void onCurrentProfileChanged(Profile oldProfile, Profile newProfile) {
        nextActivity(newProfile);
    }
};
accessTokenTracker.startTracking();
profileTracker.startTracking();

LoginButton loginButton = (LoginButton)findViewById(R.id.login_button);
callback = new FacebookCallback<LoginResult>() {
    @Override
    public void onSuccess(LoginResult loginResult) {
        AccessToken accessToken = loginResult.getAccessToken();
        Profile profile = Profile.getCurrentProfile();
        nextActivity(profile);
        Toast.makeText(getApplicationContext(), "Accediendo...", Toast.LENGTH_SHORT).show();
    }

    @Override
    public void onCancel() {
    }

    @Override
    public void onError(FacebookException e) {
    }
};
loginButton.setReadPermissions("user_friends");
loginButton.registerCallback(callbackManager, callback);
}

```

```

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Agrega items al action bar
    getMenuInflater().inflate(R.menu.menu_login, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {

    int id = item.getItemId();

    if (id == R.id.action_settings) {
        return true;
    }

    return super.onOptionsItemSelected(item);
}

@Override
protected void onResume() {
    super.onResume();
    //Facebook login
    Profile profile = Profile.getCurrentProfile();
    nextActivity(profile);
}

```

```

@Override
protected void onPause() {

    super.onPause();

}

protected void onStop() {
    super.onStop();
    //Facebook login
    accessTokenTracker.stopTracking();
    profileTracker.stopTracking();
}

@Override
protected void onActivityResult(int requestCode, int responseCode, Intent intent) {
    super.onActivityResult(requestCode, responseCode, intent);
    //Facebook login
    callbackManager.onActivityResult(requestCode, responseCode, intent);
}

private void nextActivity(Profile profile){
    if(profile != null){
        //Muestra los datos del usuario de Facebook
        Intent main = new Intent(LoginActivity.this, MainActivity.class);
        main.putExtra("name", profile.getFirstName());
        main.putExtra("surname", profile.getLastName());
        main.putExtra("imageUrl", profile.getProfilePictureUri(200,200).toString());
        startActivity(main);
    }
}

public void printKeyHash() {
    //Funcion para imprimir el key hash
    try {
        PACKAGE_NAME = getApplicationContext().getPackageName();
        PackageInfo info = getPackageManager().getPackageInfo(PACKAGE_NAME, PackageManager.GET_SIGNATURES);
        for (Signature signature : info.signatures) {
            MessageDigest md = MessageDigest.getInstance("SHA");
            md.update(signature.toByteArray());
            Log.d("Key Hash -->", Base64.encodeToString(md.digest(), Base64.DEFAULT));
        }
    } catch (PackageManager.NameNotFoundException e) {

    } catch (NoSuchAlgorithmException e) {

    }
}
}

```

6. De la misma manera complete el MainActivity

```

public class MainActivity extends AppCompatActivity {

    private ShareDialog mShareDialog;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        FacebookSdk.sdkInitialize(getApplicationContext());
        setContentView(R.layout.activity_main);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        Bundle inBundle = getIntent().getExtras();
        String name = inBundle.get("name").toString();
        String surname = inBundle.get("surname").toString();
        String imageUrl = inBundle.get("imageUrl").toString();

        TextView nameView = (TextView) findViewById(R.id.nameAndSurname);
        nameView.setText(" "+name+" "+surname);

        mShareDialog = new ShareDialog(this);
        new DownloadImage((ImageView) findViewById(R.id.profileImage)).execute(imageUrl);

        FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                ShareLinkContent content = new ShareLinkContent.Builder().build();
                mShareDialog.show(content);
            }
        });
    }
}

```

```

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Agregar items al action bar
        getMenuInflater().inflate(R.menu.menu_login, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.action_settings) {
            logout();
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    public void logout(){
        //Funcion para salir
        LoginManager.getInstance().logout();
        Intent login = new Intent(MainActivity.this, LoginActivity.class);
        startActivity(login);
        finish();
    }
}

```

Pruebe su aplicación y guarde los screenshots.

Preguntas

1. ¿Qué es y para qué sirve LoginManager?
2. ¿Cuál es el API level mínimo para usar el sdk de Facebook?
3. ¿Para qué sirve la clase CallbackManager?
4. ¿Qué sucede si intenta hacer login sin haber hecho la aplicación pública?
5. Tome un screenshot de la aplicación funcionando y publíquelo en el grupo de Facebook del curso desde la misma aplicación junto con el nombre de los integrantes.

Bibliografía

- Facebook. (25 de Abril de 2016). *facebook for developers*. Obtenido de <https://developers.facebook.com/docs/facebook-login/android>
- Facebook. (25 de Abril de 2016). *facebook for developers*. Obtenido de <https://developers.facebook.com/docs/facebook-login/android>
- Facebook. (27 de Abril de 2016). *facebook for developers*. Obtenido de <https://developers.facebook.com/docs/reference/android/current/interface/CallbackManager/>
- Ortiz, C. E. (19 de Diciembre de 2010). *ibm*. Obtenido de <http://www.ibm.com/developerworks/library/x-androidfacebookapi/>
- Xamarin Inc. (7 de Abril de 2016). *Xamarin*. Obtenido de <https://components.xamarin.com/gettingstarted/facebookandroid>