1. **Historial de cambios**

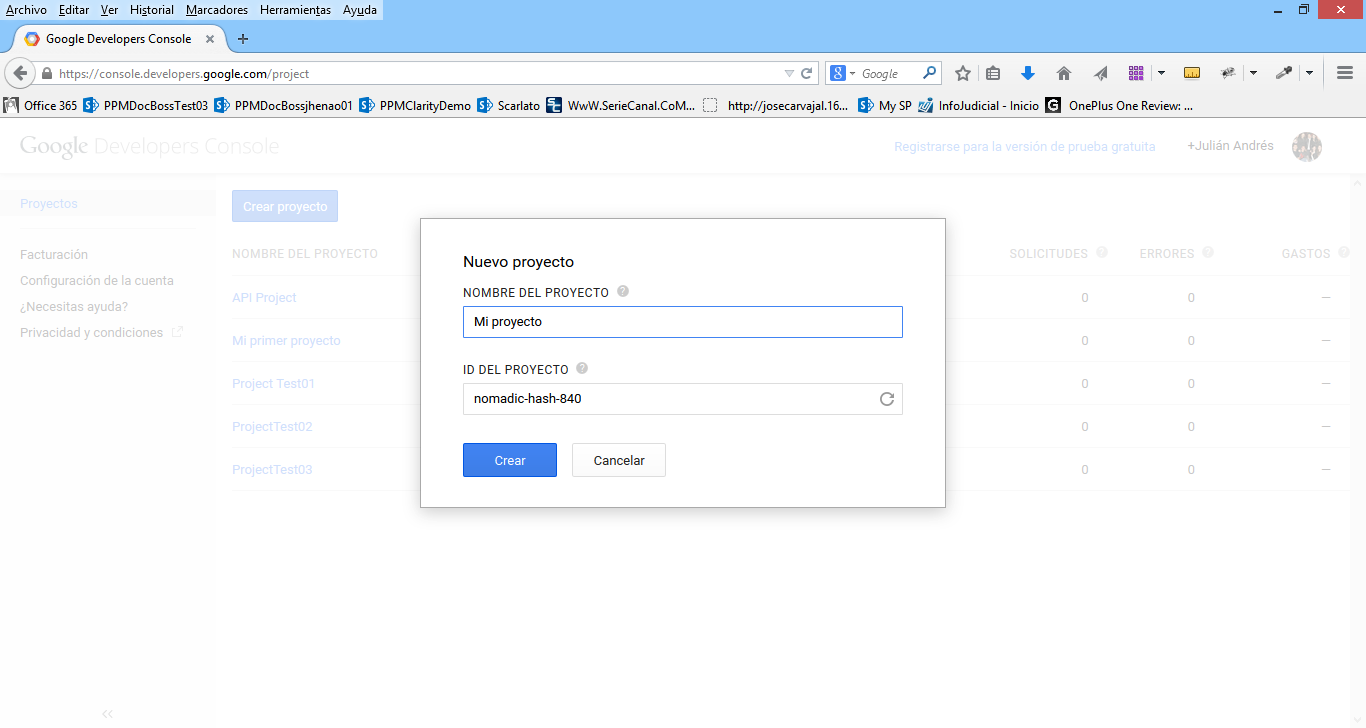
| **Fecha** | **Versión** | **Autor** | **Descripción del Cambio** |
| --- | --- | --- | --- |
| 22/03/2015 | 1.0 | Julián Andrés Henao Taborda | Creación del documento |

1. **Introducción**

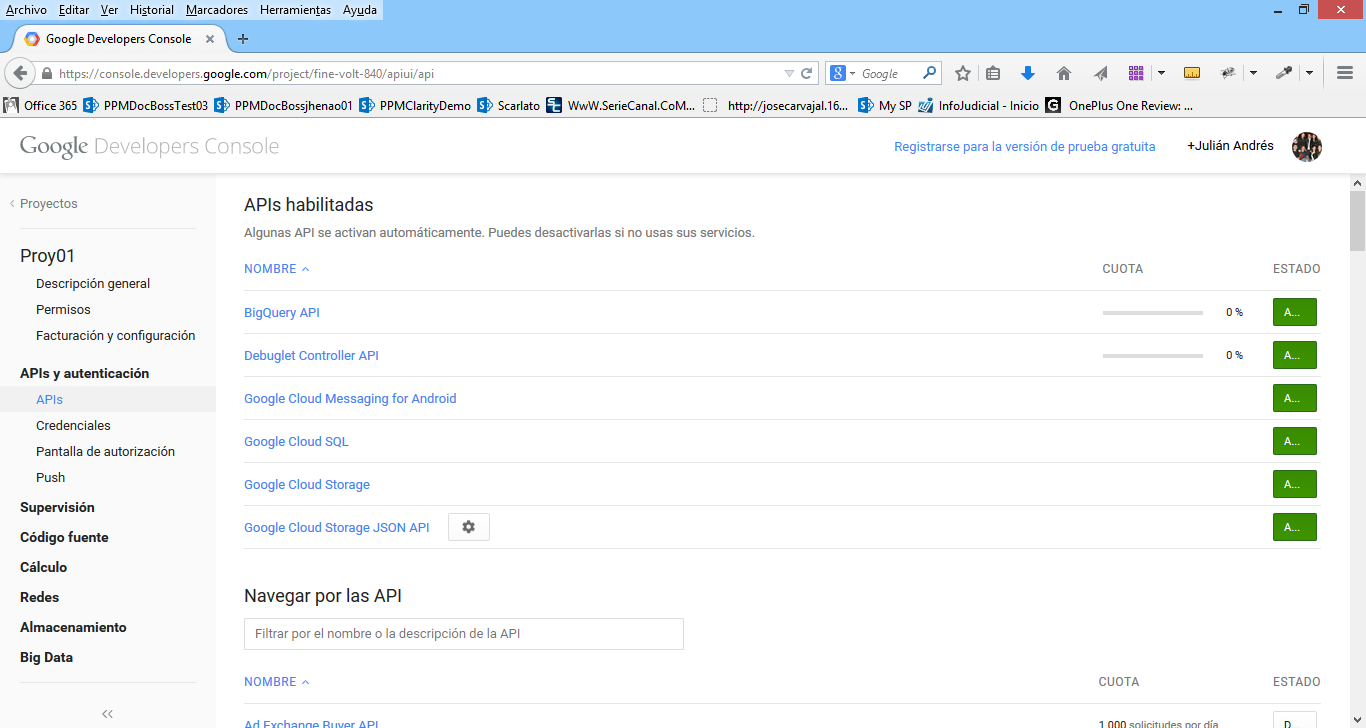
El objetivo del tutorial es el desarrollo de una aplicación cliente para sistema operativo Android para hacer uso de *Google Cloud Messaging.*

Después de seguir este tutorial se obtendrá una aplicación que una vez instalada en un dispositivo podrá registrarse al servidor *Google Cloud Messaging* y así poder recibir mensajes desde una aplicación servidor.

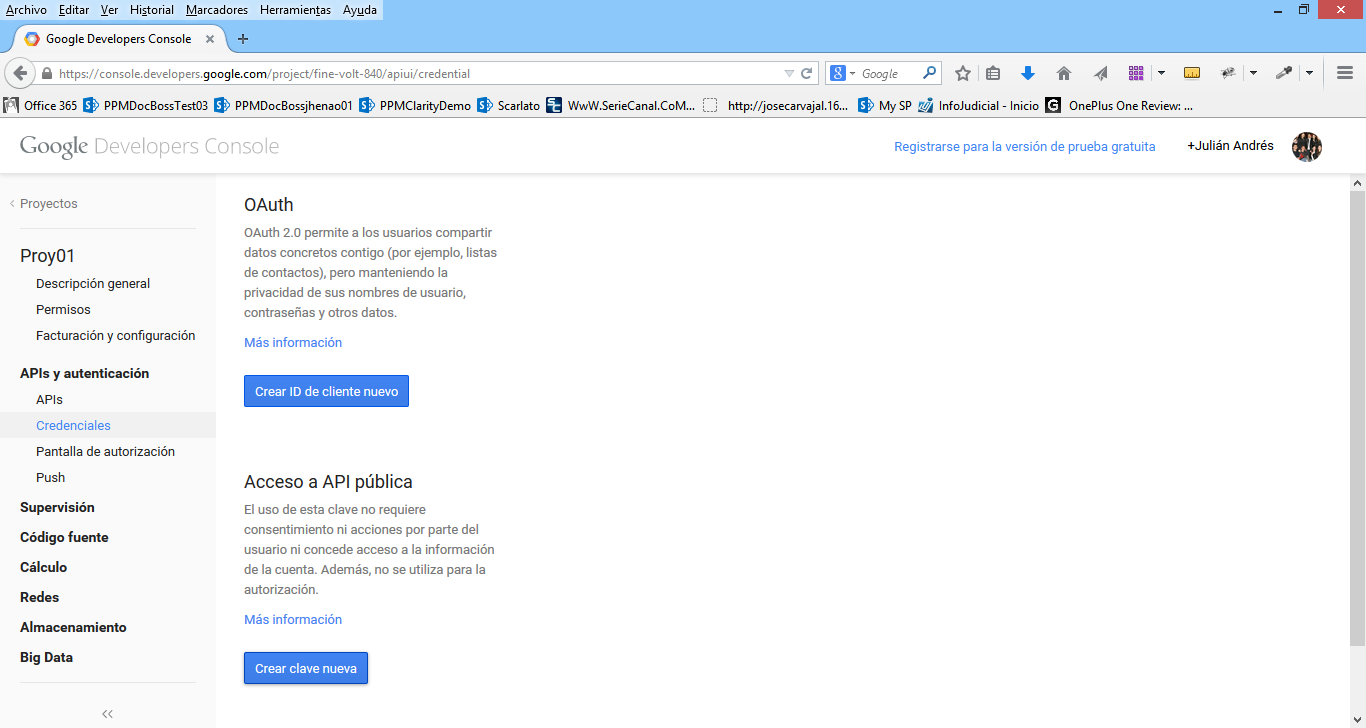
1. Antes de proceder con la instalación es necesario registrar una cuenta de Google en la Consola de Google (<https://console.developers.google.com/project/>).
   1. Crear un proyecto



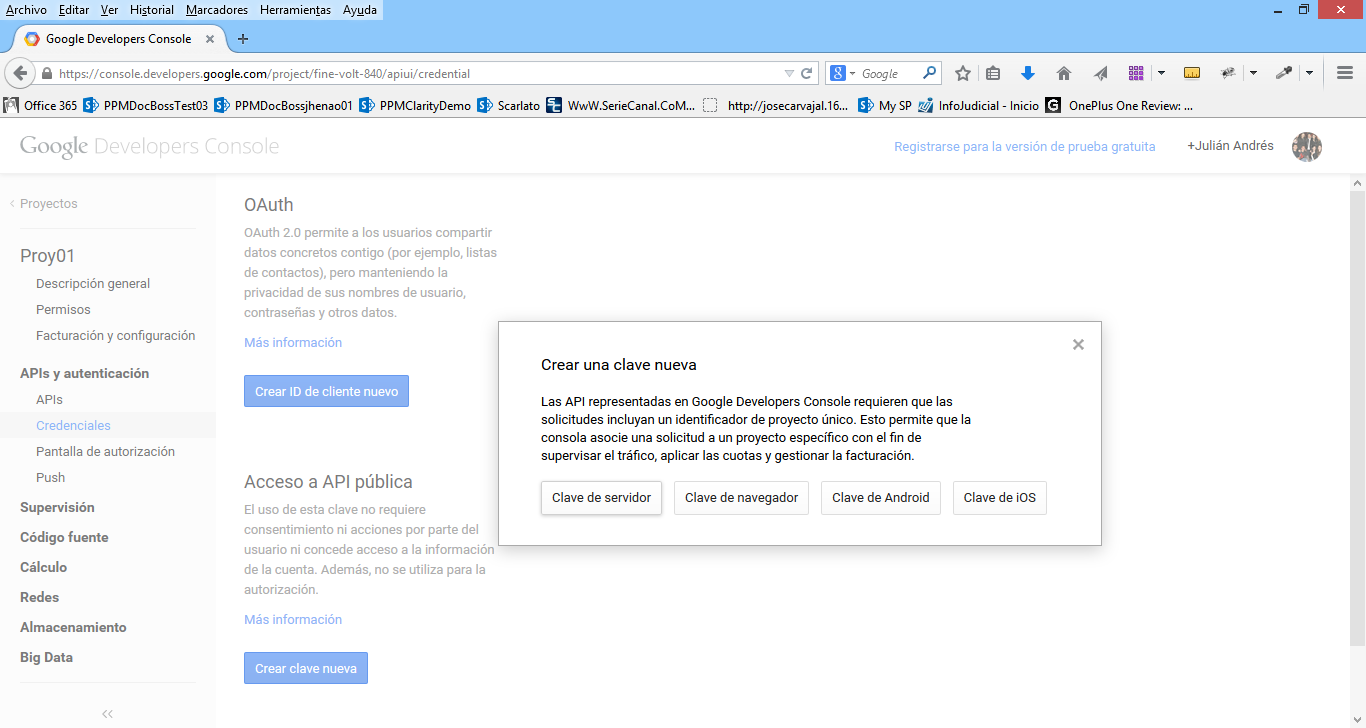
* 1. Acceder al proyecto y en “APISs y autenticación” seleccionar “APIs” y habilitar “Google Cloud Messaging for Android”.

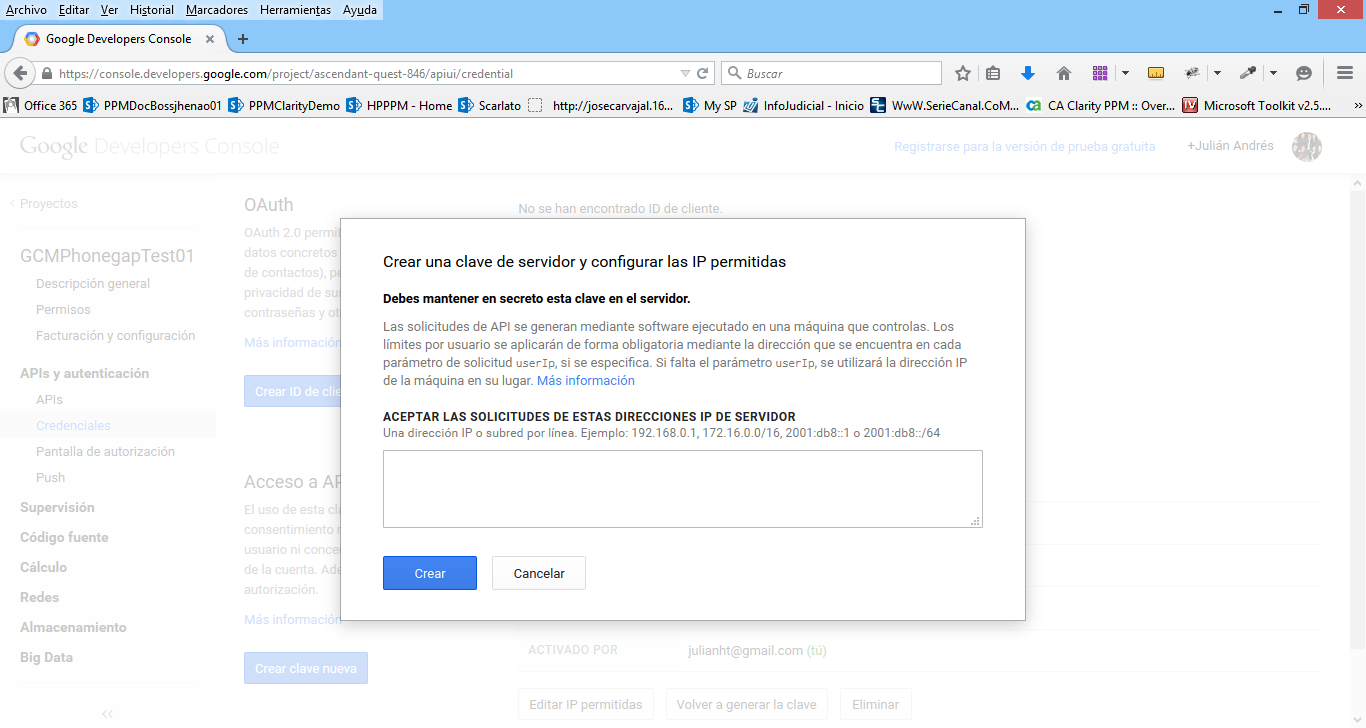


* 1. En “APIs y autenticación” seleccionar “Credenciales” y “Crear clave nueva”.

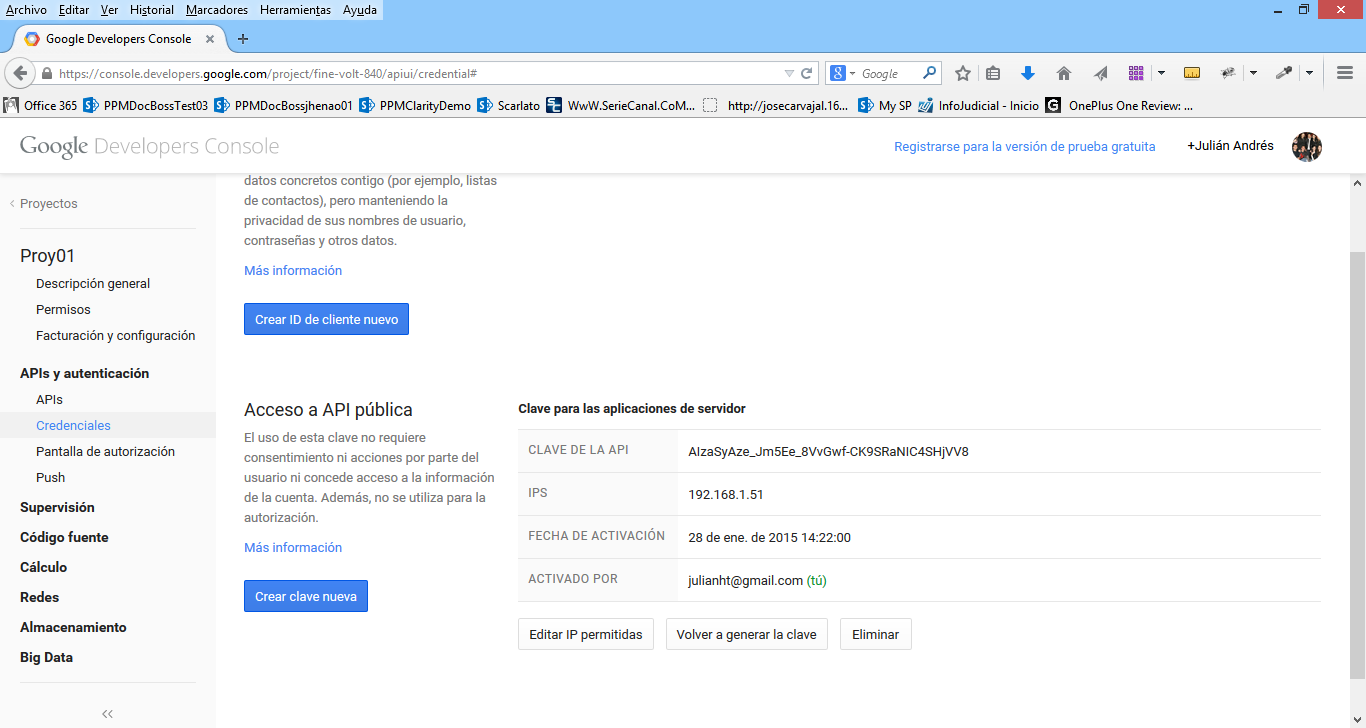


* 1. Seleccionar “Clave servidor” y “Crear”.

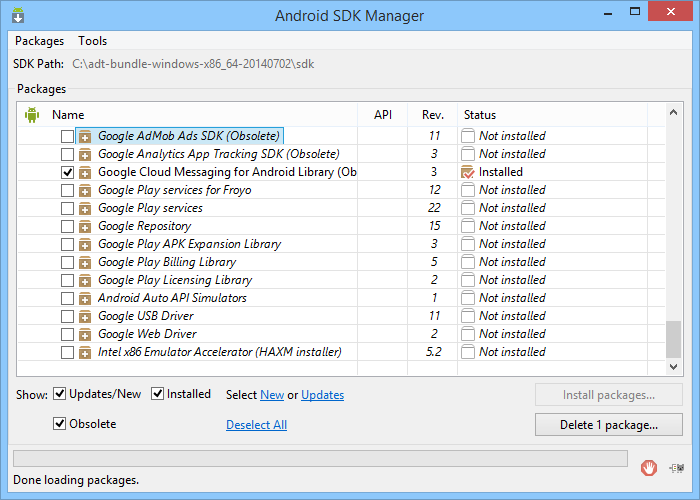




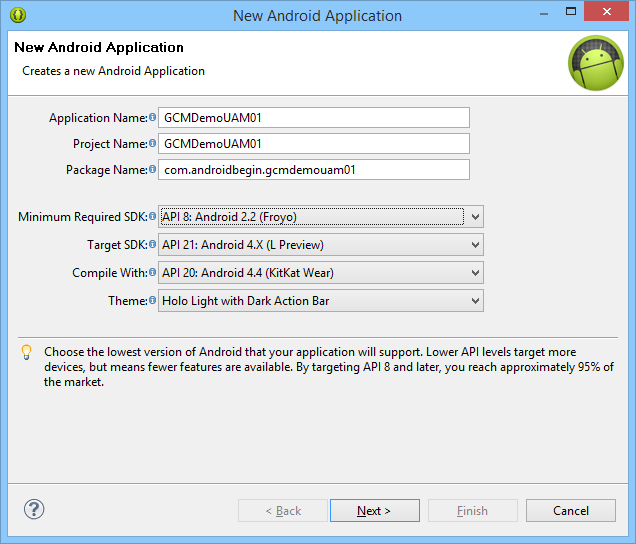
* 1. Una vez creada copiar el código definido en Clave de la API (Será usado más adelante).

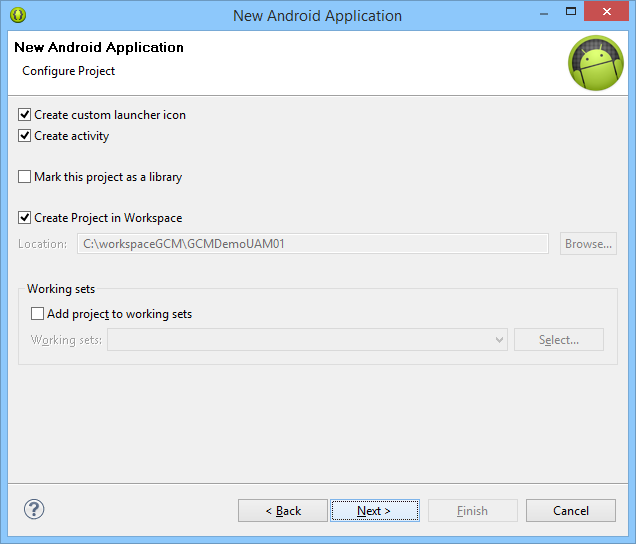


1. Descargar Eclipse SDK.
2. Ir al Android SDK Manager y descargar “Google Cloud Messaging for Android Library”

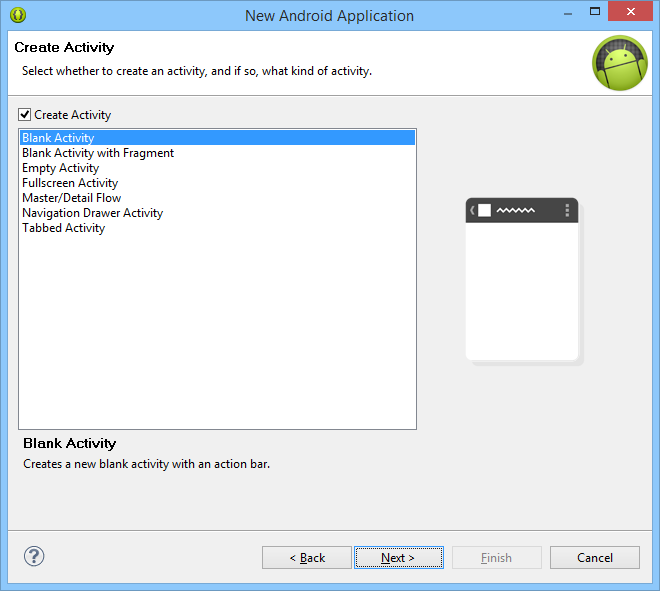


1. En Eclipse crear un nuevo proyecto tipo -> Android Application Project.





1. Crear “Blank Activity”



1. Copiar el archivo “gcm.jar” ubicado en el directorio SKD donde se instaló “Google Cloud Messaging for Android Library” (sdk\extras\google\gcm\samples\gcm-demo-client\libs\gcm.jar) y pegarlo en la carpeta “libs” del proyecto recientemente creado.
2. Reemplazar el contenido de la clase creada por el siguiente código

**En esta clase se programan los eventos de los botones de registro del dispositivo, así como la consulta del id generado para el dispositivo que posteriormente será utilizado para el envío de los mensajes desde la aplicación servidor. También se definen validaciones para el registro y la consulta del mismo.**

**package** com.androidbegin.gcmdemouam01;

**import** com.google.android.gcm.GCMRegistrar;

**import** android.os.Bundle;

**import** android.app.Activity;

**import** android.util.Log;

**import** android.view.View;

**import** android.widget.Button;

**import** android.widget.TextView;

**import** android.app.AlertDialog;

**import** android.content.DialogInterface;

**public** **class** GCMMainActivity **extends** Activity {

//Encabezado de la aplicación para el registro de logs

String TAG = "GCM DEMO UAM";

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_main*);

// Se realizan las validaciones de la aplicación en el servicio GCM

GCMRegistrar.*checkDevice*(**this**);

GCMRegistrar.*checkManifest*(**this**);

// Botón para registrar el dispositivo

Button regbtn = (Button) findViewById(R.id.*register*);

regbtn.setOnClickListener(**new** View.OnClickListener() {

@Override

**public** **void** onClick(View v) {

String regId = GCMRegistrar.*getRegistrationId*(GCMMainActivity.**this**);

AlertDialog alertDialog = **new** AlertDialog.Builder(GCMMainActivity.**this**).create();

**if**(regId.equals(""))

{

//Se obtiene el senderID para registrarlo

GCMRegistrar.*register*(GCMMainActivity.**this**,

GCMIntentService.*SENDER\_ID*);

alertDialog.setMessage("Registrando dispositivo");

Log.*i*(TAG, "Registrando dispositivo");

}

**else**

{

alertDialog.setMessage("El dispositivo ya se encuentra registrado");

Log.*i*(TAG, "El dispositivo ya se encuentra registrado");

}

alertDialog.setButton("OK", **new** DialogInterface.OnClickListener() {

**public** **void** onClick(DialogInterface dialog, **int** which) {

}

});

alertDialog.show();

}

});

//Botón para la consulta del id del dispositivo

Button btnConsultarId = (Button)findViewById(R.id.*consultarIdRegistro*);

btnConsultarId.setOnClickListener(**new** View.OnClickListener() {

@Override

**public** **void** onClick(View v) {

String regId = GCMRegistrar.*getRegistrationId*(GCMMainActivity.**this**);

**if**(regId.equals(""))

{

AlertDialog alertDialog = **new** AlertDialog.Builder(GCMMainActivity.**this**).create();

alertDialog.setMessage("Dispositivo no registrado");

alertDialog.setButton("OK", **new** DialogInterface.OnClickListener() {

**public** **void** onClick(DialogInterface dialog, **int** which) {

}

});

alertDialog.show();

}

**else**

{

TextView txtIdRegistro = (TextView)findViewById(R.id.*registerid*);

txtIdRegistro.setText(regId);

}

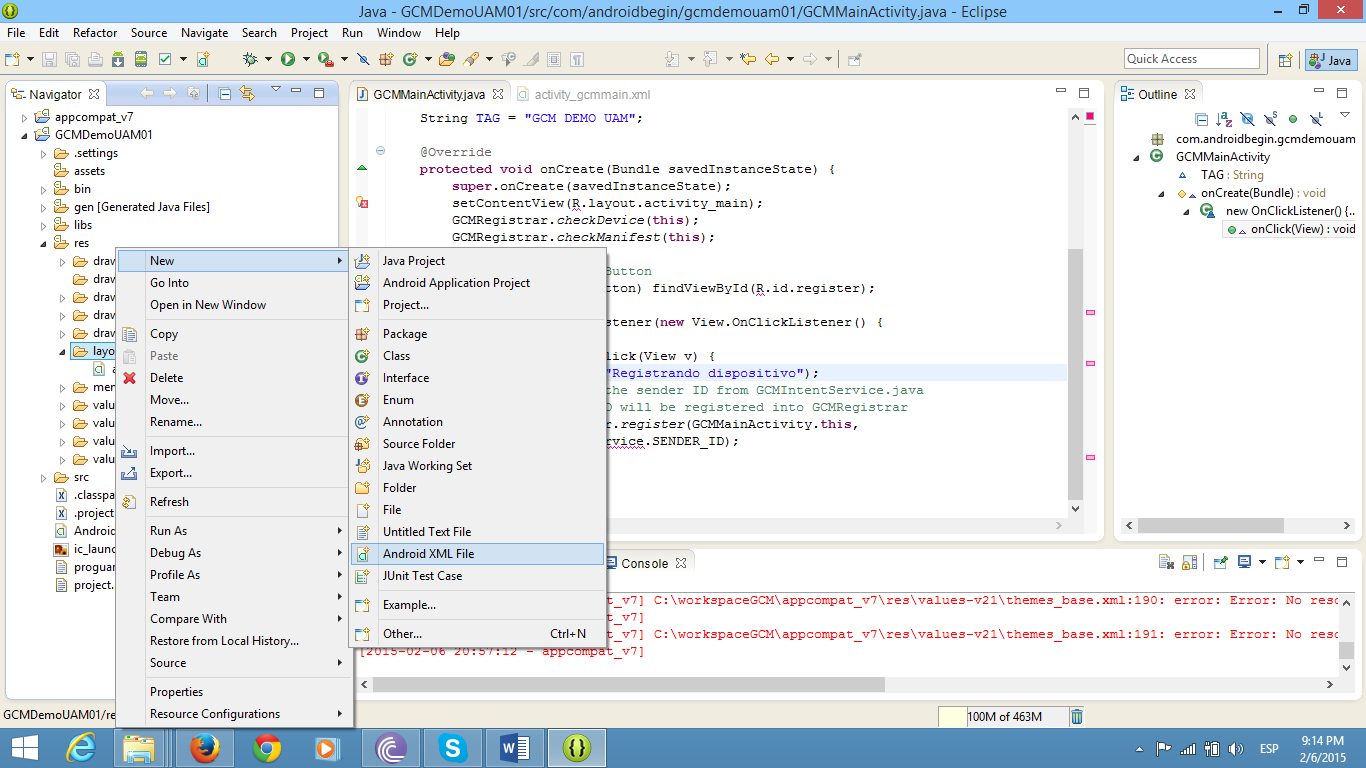
}

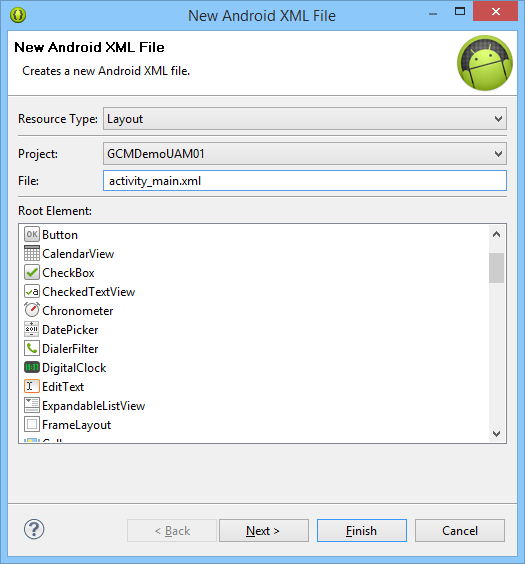
});

}

}

1. Proceder a crear la plantilla gráfica para la clase. Creando un “Android XML File” en el directorio de layouts:





1. Reemplazar el contenido del XML creado por el código:

**En el xml se definen los controles android de la interfaz gráfica del formulario de registro de la aplicación.**

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" >

<Button

android:id="@+id/register"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="80dp"

android:text="@string/regbutton" />

<Button

android:id="@+id/consultarIdRegistro"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/register"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="32dp"

android:text="@string/consultarbutton" />

<TextView

android:id="@+id/registerid"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/consultarIdRegistro"

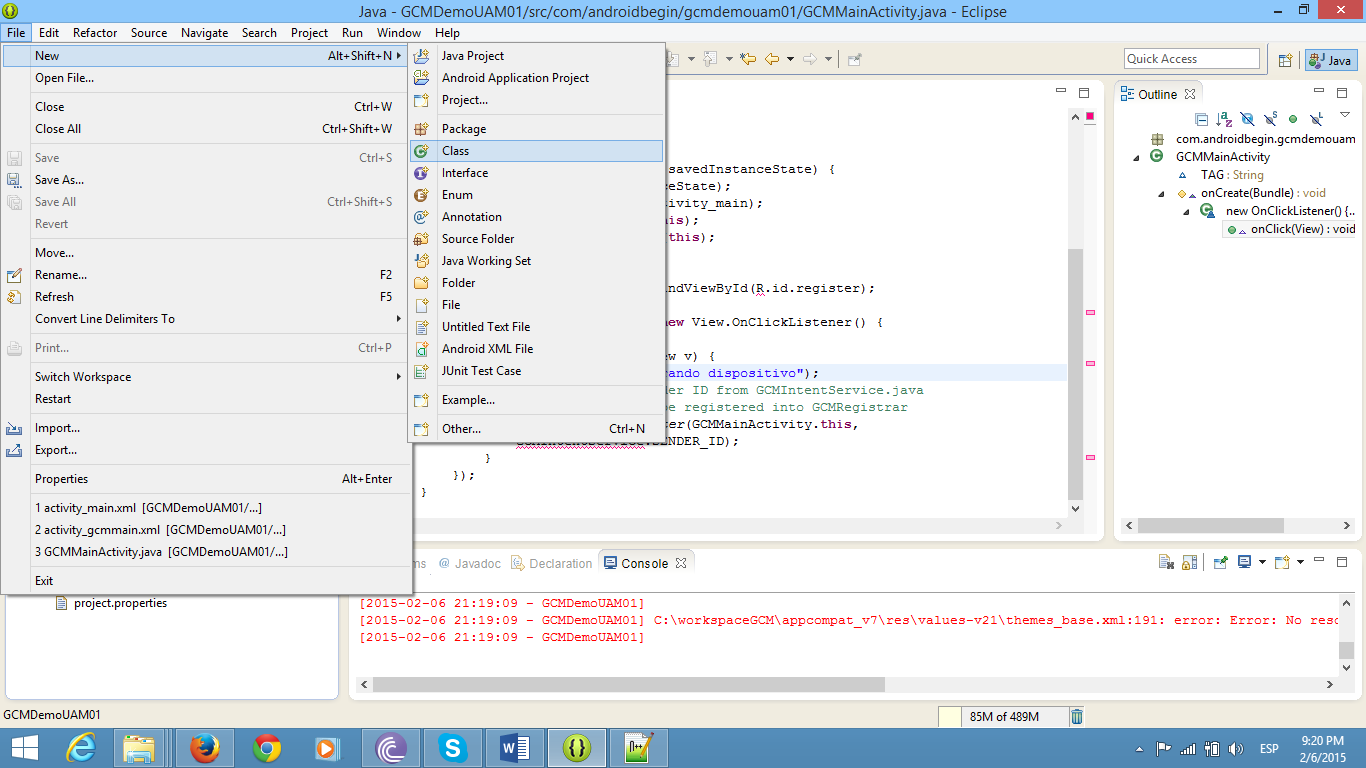
android:layout\_centerHorizontal="true"

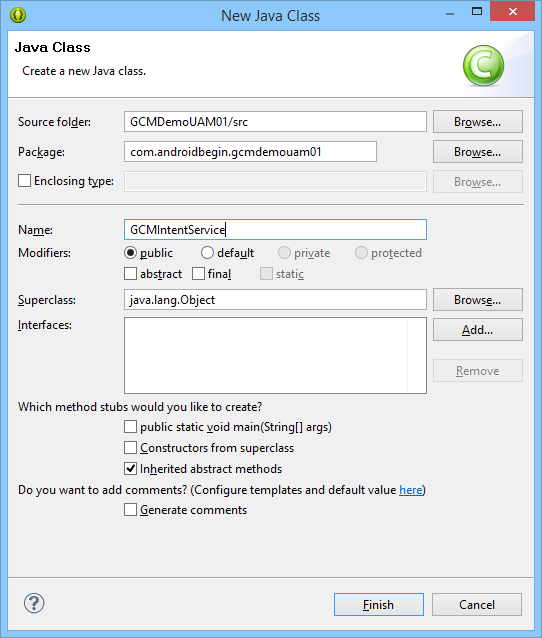
android:textIsSelectable="true"

android:layout\_marginTop="36dp" />

</RelativeLayout>

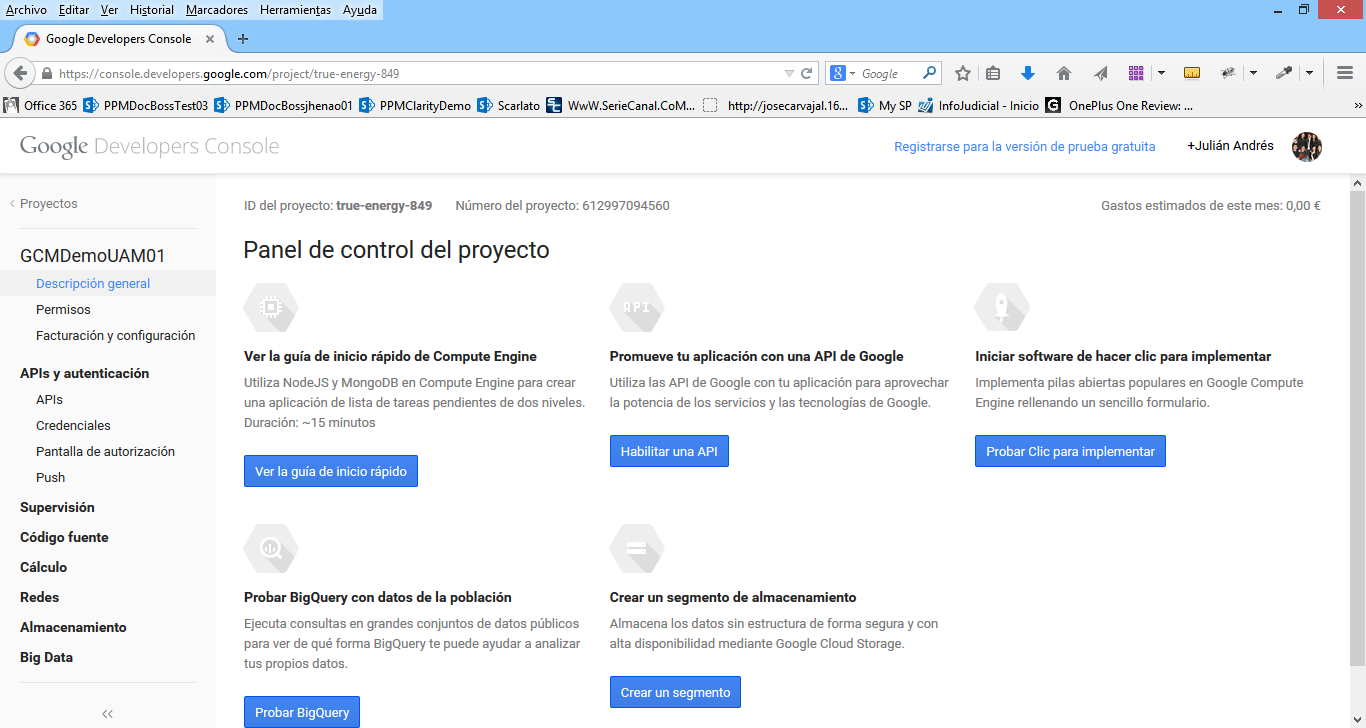
1. Crear una nueva clase en el proyecto:





1. Reemplazar el contenido de la clase creada por el código:

Reemplazar el valor de la variable SENDER\_ID por el ID del proyecto creado en la consola de Google:



**En esta clase se definen los servicios de notificación con el dispositivo al recibir un mensaje desde el servicio GCM (Para nuestro caso sería la aplicación servidor).**

**package** com.androidbegin.gcmdemouam01;

**import** java.util.Timer;

**import** java.util.TimerTask;

**import** android.app.Notification;

**import** android.app.NotificationManager;

**import** android.app.PendingIntent;

**import** android.content.Context;

**import** android.content.Intent;

**import** android.os.PowerManager;

**import** android.util.Log;

**import** com.google.android.gcm.GCMBaseIntentService;

**public** **class** GCMIntentService **extends** GCMBaseIntentService {

**private** **static** **final** String *TAG* = "GCM DEMO UAM";

//Reemplazar el valor de la variable SENDER\_ID por el número del

//proyecto creado en la consola de Google

**public** **static** **final** String *SENDER\_ID* = "612997094560";

**public** GCMIntentService() {

**super**(*SENDER\_ID*);

}

@Override

**protected** **void** onRegistered(Context context,

String registrationId) {

Log.*i*(*TAG*, "onRegistered: registrationId=" + registrationId);

}

@Override

**protected** **void** onUnregistered(Context context,

String registrationId) {

Log.*i*(*TAG*, "onUnregistered: registrationId=" + registrationId);

}

@Override

**protected** **void** onMessage(Context context, Intent data) {

String message;

//Mensaje obtenido de la aplicación servidor

message = data.getStringExtra("message");

// Open a new activity called GCMMessageView

Intent intent = **new** Intent(**this**, GCMMessageView.**class**);

// Pass data to the new activity

intent.putExtra("message", message);

// Starts the activity on notification click

PendingIntent pIntent = PendingIntent.*getActivity*(**this**, 0, intent, PendingIntent.*FLAG\_UPDATE\_CURRENT*);

//Create the notification with a notification builder

Notification notification = **new** Notification.Builder(**this**)

.setSmallIcon(R.drawable.ic\_launcher)

.setWhen(System.*currentTimeMillis*())

.setContentTitle("Android GCM Tutorial")

.setContentText(message)

.setContentIntent(pIntent)

.getNotification();

// Remove the notification on click

notification.flags |= Notification.*FLAG\_AUTO\_CANCEL*;

NotificationManager manager = (NotificationManager)getSystemService(*NOTIFICATION\_SERVICE*);

manager.notify(R.string.app\_name, notification);

{

// Wake Android Device when notification received

PowerManager pm = (PowerManager)context.getSystemService(Context.*POWER\_SERVICE*);

**final** PowerManager.WakeLock mWakelock = pm.newWakeLock(PowerManager.~~FULL\_WAKE\_LOCK~~ | PowerManager.*ACQUIRE\_CAUSES\_WAKEUP*, "GCM\_PUSH");

mWakelock.acquire();

// Timer before putting Android Device to sleep mode.

Timer timer = **new** Timer();

TimerTask task = **new** TimerTask() {

**public** **void** run() {

mWakelock.release();

}

};

timer.schedule(task, 5000);

}

}

@Override

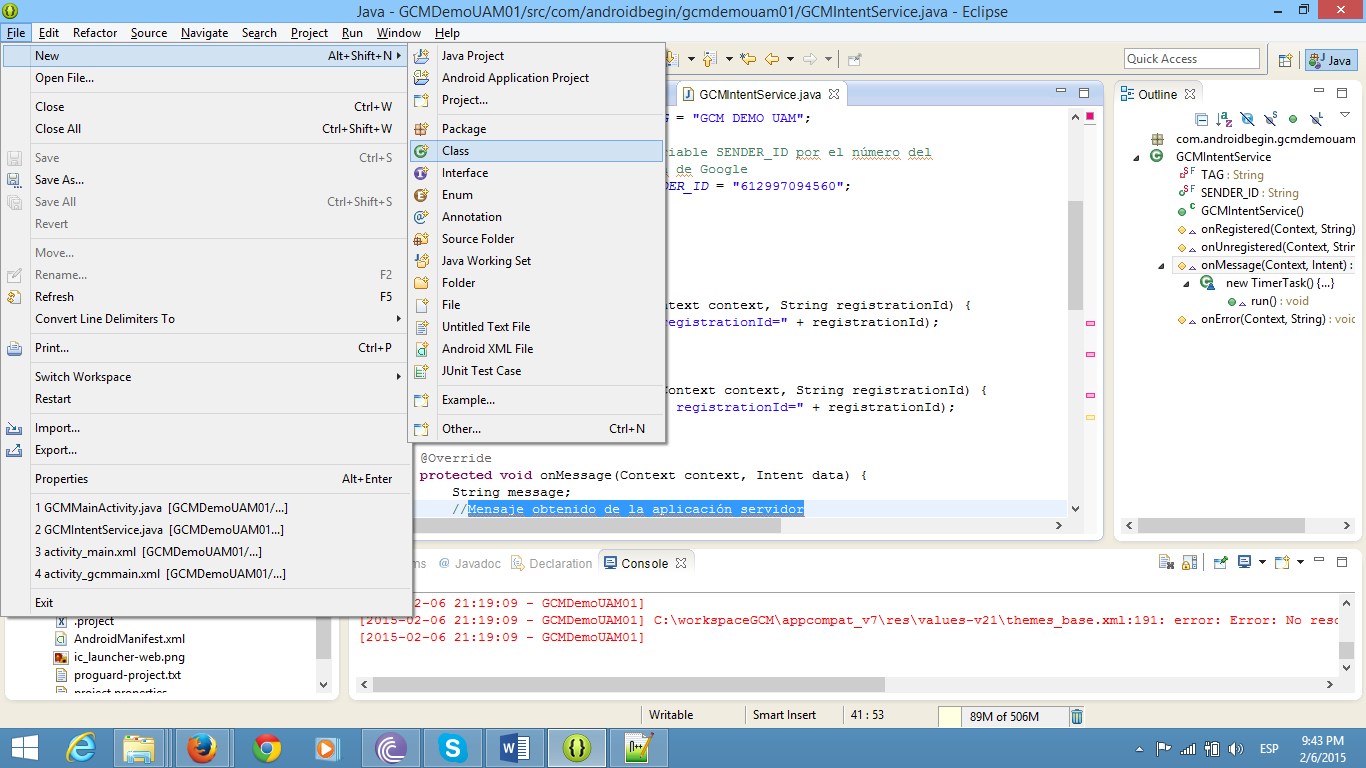
**protected** **void** onError(Context arg0, String errorId) {

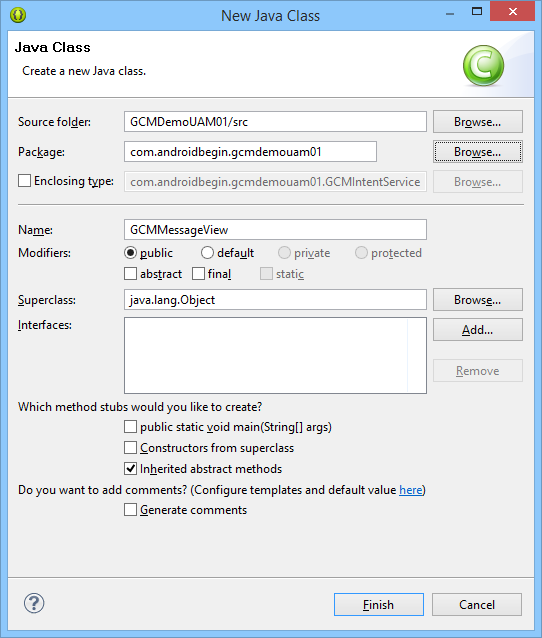
Log.*e*(*TAG*, "onError: errorId=" + errorId);

}

}

1. Crear una clase para mostrar el mensaje de notificación:





1. Reemplazar el contenido de la clase creada por el código:

**Esta clase define la interfaz gráfica mediante la cual se muestra el mensaje de notificación en el dispositivo recibido desde el servicio GCM.**

**package** com.androidbegin.gcmdemouam01;

**import** android.app.Activity;

**import** android.content.Intent;

**import** android.os.Bundle;

**import** android.widget.TextView;

**public** **class** GCMMessageView **extends** Activity {

String message;

TextView txtmsg;

@Override

**public** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.messageview);

//Retrive the data from GCMIntentService.java

Intent i = getIntent();

message = i.getStringExtra("message");

//Locate the TextView

txtmsg = (TextView) findViewById(R.id.message);

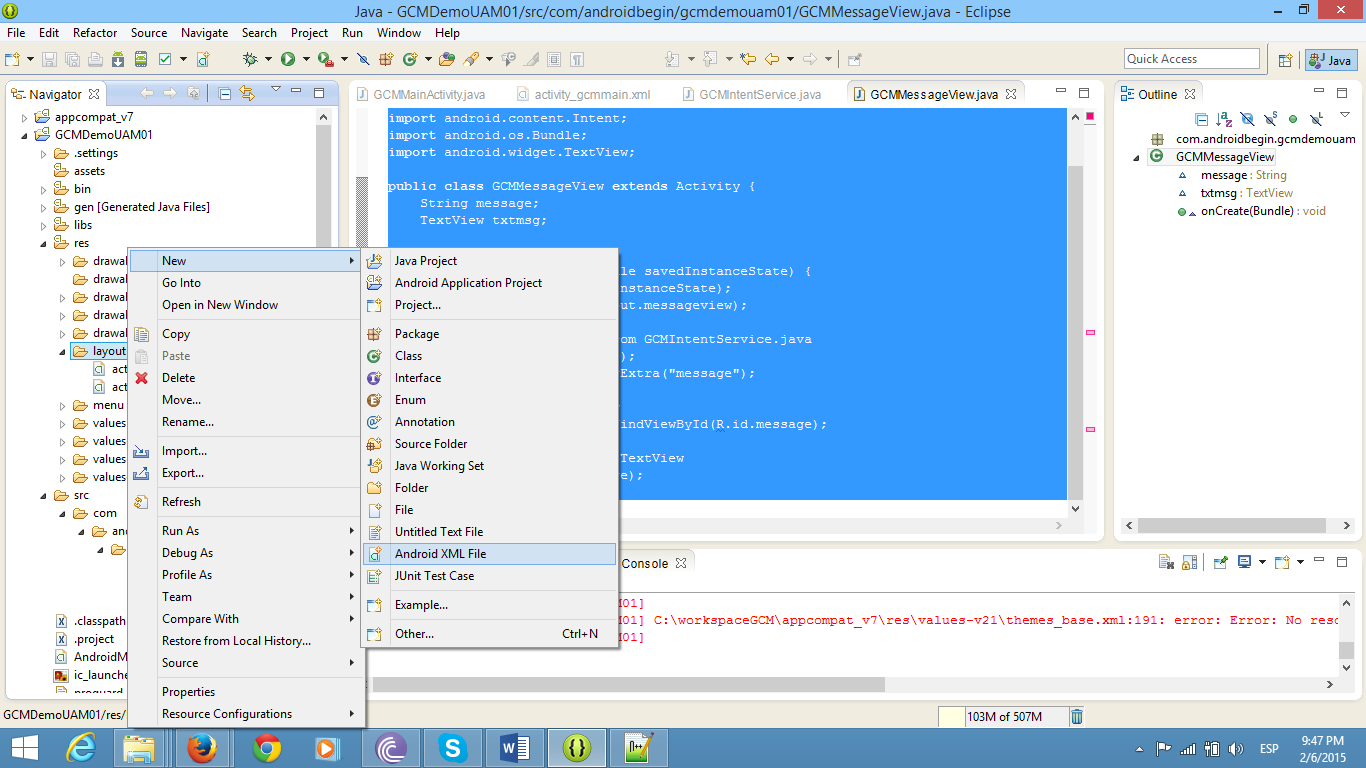
// Set the data into TextView

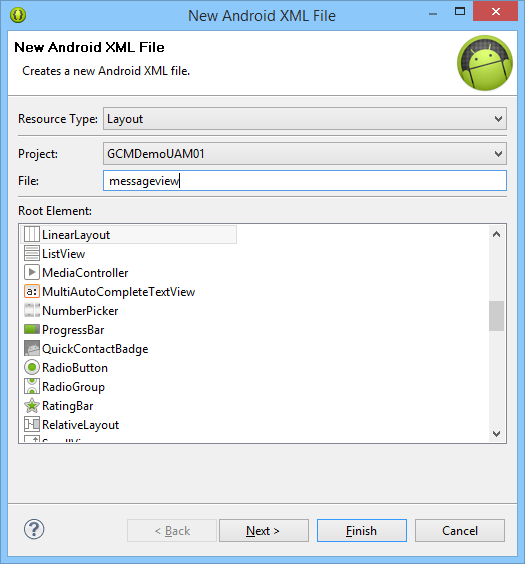
txtmsg.setText(message);

}

}

1. Proceder a crear la plantilla gráfica para la clase. Creando un “Android XML File” en el directorio de layouts:





1. Reemplazar el contenido del XML creado por el código:

**En el xml se definen los controles android de la interfaz gráfica del formulario de visualización del mensaje de la aplicación.**

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent" >

<TextView

android:id="@+id/msglbl"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/msglbl" />

<TextView

android:id="@+id/message"

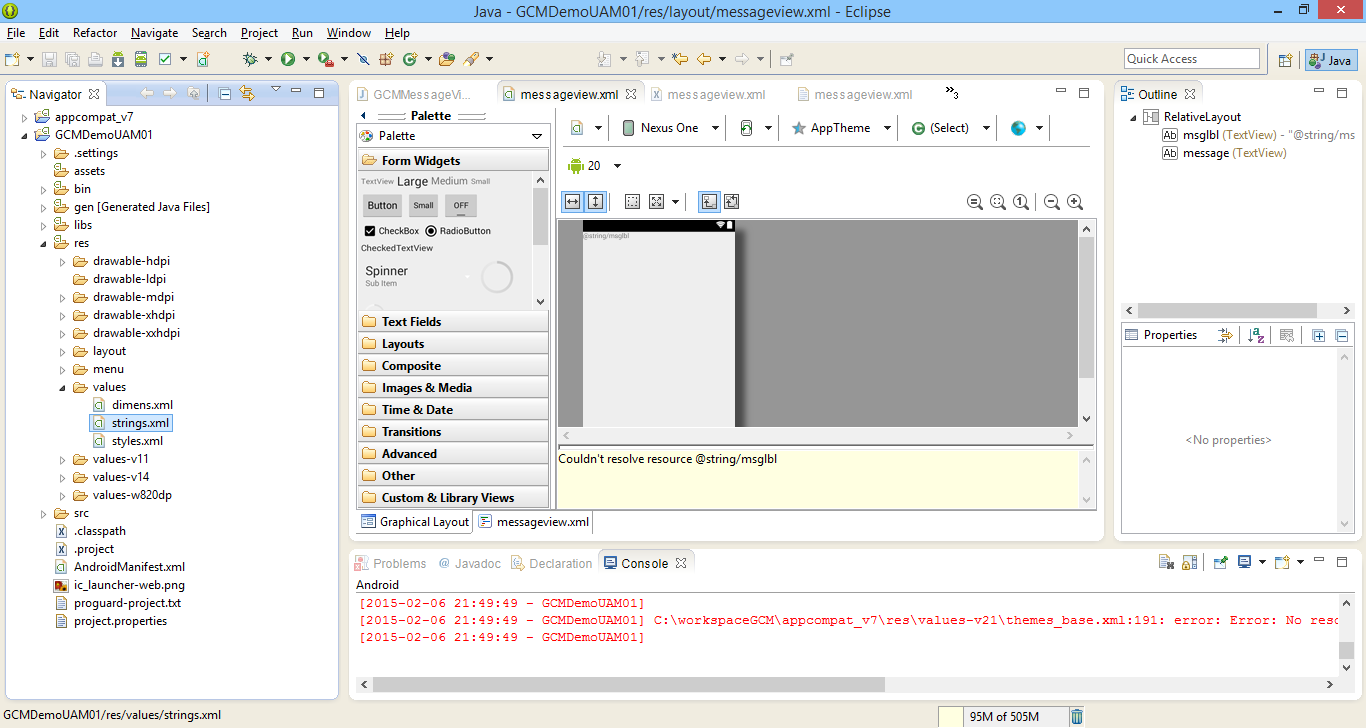
android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/msglbl" />

</RelativeLayout>

1. Procedemos a modificar las constantes con los textos de la aplicación reemplazando el contenido del archivo “**string.xml**”:



1. Reemplazamos el contenido del archivo XML por el código:

**En el xml se definen los mensajes constantes usados en la interfaz gráfica de la aplicación.**

<?xml version="1.0" encoding="utf-8"?>

<resources>

<string name="app\_name">GCM DEMO UAM</string>

<string name="hello\_world">Hola Dispositivo!</string>

<string name="menu\_settings">Configuración</string>

<string name="regbutton">Registrar Dispositivo</string>

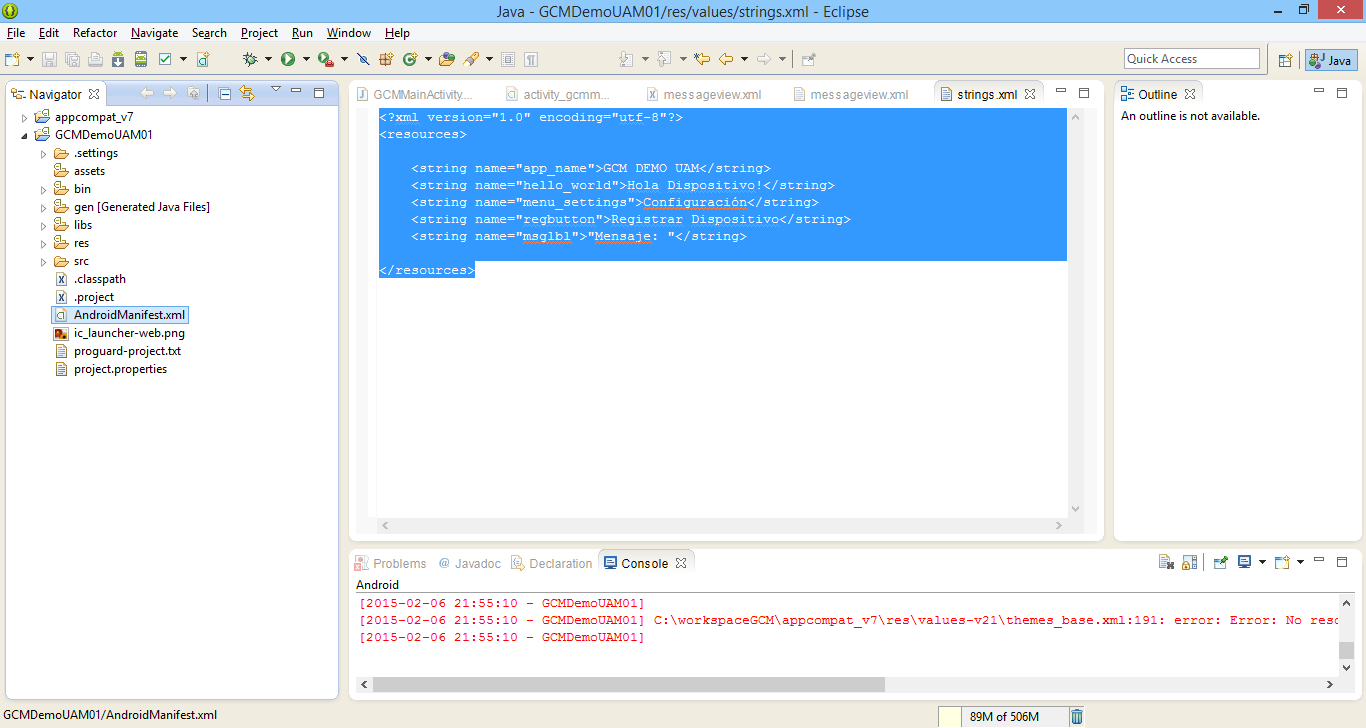
<string name="msglbl">"Mensaje: "</string>

<string name="action\_settings">Configuración</string>

<string name="consultarbutton">Ver Id Dispositivo</string>

</resources>

1. Se deben declarar los permisos necesarios para comunicar la aplicación con GCM y con el servidor web modificando el archivo “**AndroidManifest.xml**”.



1. Reemplazamos el contenido del archivo XML por el código:

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.androidbegin.gcmdemouam01"

android:versionCode="1"

android:versionName="1.0" >

<uses-sdk

android:minSdkVersion="11"

android:targetSdkVersion="15" />

<permission android:name="com.androidbegin.gcmdemouam01.permission.C2D\_MESSAGE" android:protectionLevel="signature" />

<uses-permission android:name="com.androidbegin.gcmdemouam01.permission.C2D\_MESSAGE" />

<uses-permission android:name="com.google.android.c2dm.permission.RECEIVE" />

<uses-permission android:name="android.permission.WAKE\_LOCK" />

<application

android:allowBackup="false"

android:icon="@drawable/ic\_launcher"

android:label="@string/app\_name" >

<activity

android:name=".GCMMainActivity"

android:launchMode="singleTask" >

<intent-filter>

<action android:name="android.intent.action.MAIN" />

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

</intent-filter>

</activity>

<activity android:name=".GCMMessageView" >

</activity>

<service android:name=".GCMIntentService" />

<receiver

android:name="com.google.android.gcm.GCMBroadcastReceiver"

android:permission="com.google.android.c2dm.permission.SEND" >

<intent-filter>

<action android:name="com.google.android.c2dm.intent.RECEIVE" />

<action android:name="com.google.android.c2dm.intent.REGISTRATION" />

<category android:name="com.androidbegin.gcmdemouam01" />

</intent-filter>

</receiver>

</application>

</manifest>

1. Instalar la aplicación en el dispositivo para obtener el “**Id del dispositivo**”.



1. Copiar el id del dispositivo para pegarlo en la aplicación servidor.