



Universidad Nacional
Jorge Basadre Grohmann



PROGRAMACIÓN PARA DISPOSITIVOS MÓVILES: ANDROID Parte IV

MSc. ING. WILSON CÉSAR CALLISAYA CHOQUECOTA

TEMARIO

- Servicios
- Notificaciones
- Servicios en Android 9
- Receptores de Anuncios

Servicios



CREAMOS UN NUEVO PROYECTO CLASE4


Empty Activity

El nombre de la actividad principal será Servicio

Descargamos el siguiente mp3

<https://drive.google.com/file/d/1ffvixRxKenVNcaDakIRhjD6Jsdd0XLDp/view?usp=sharing>

Creamos la actividad Servicio
Creamos una Java Class ServicioMusica

 Create New Class ✕

Name:

Kind: Class ▼

Superclass:


Interface(s):

Package:

Visibility: ☒ Public ☐ Package Private

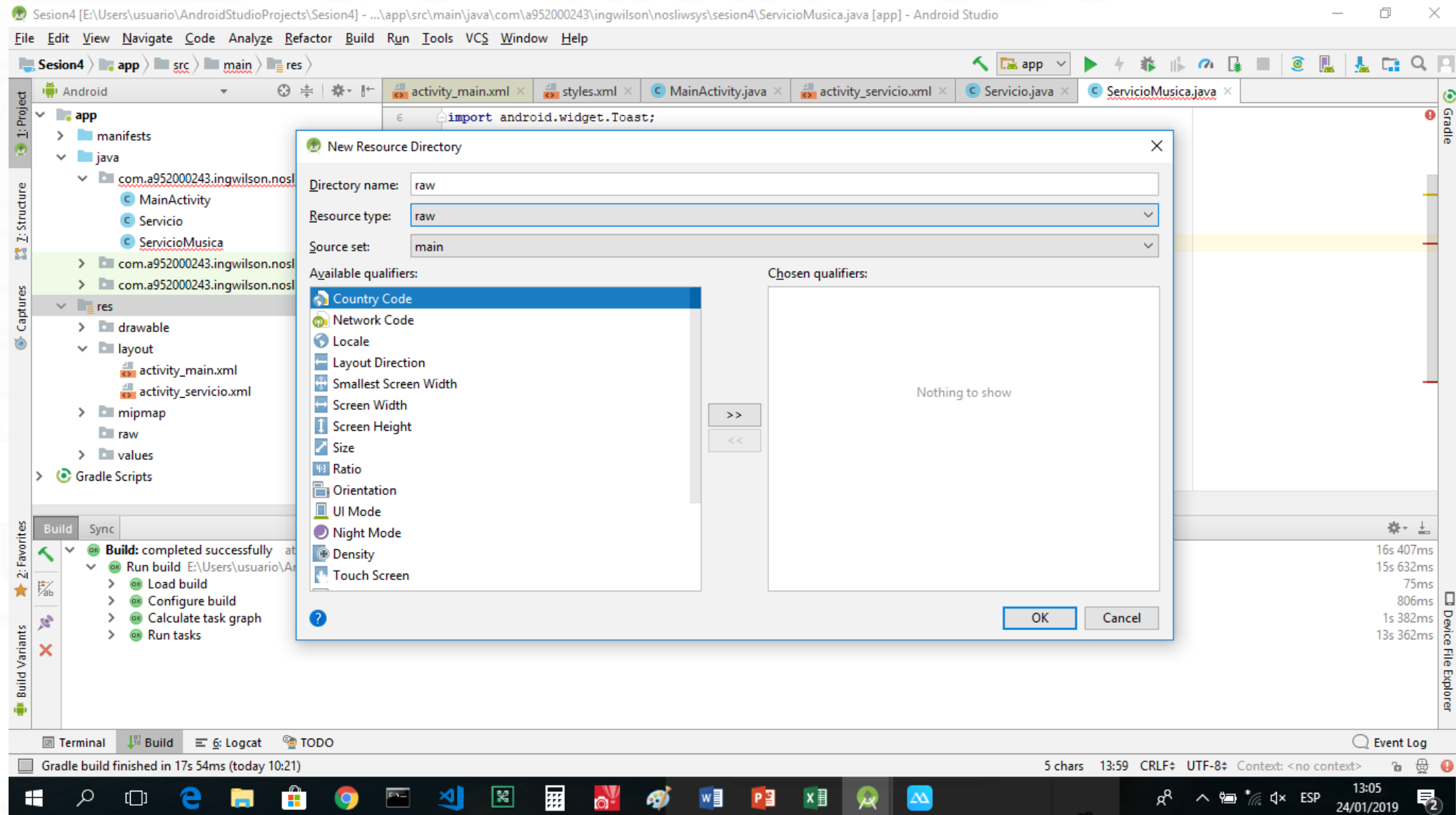
Modifiers: ☒ None ☐ Abstract ☐ Final

☐ Show Select Overrides Dialog



[illegible]

Añadimos la carpeta de recurso Raw



Copiamos y pegamos el archivo música.mp3 en raw

The screenshot shows the Android Studio interface with the `Sesion4` project open. The `raw` directory under `res` is selected in the Project view. The `ServicioMusica.java` file is open in the editor, showing the `onCreate()` method where a `MediaPlayer` is created from `R.raw.audio`. A "Copy" dialog box is displayed, showing the source file `C:\Users\TOSHIBA\Downloads\musica.mp3` and the destination directory `E:\Users\usuario\AndroidStudioProjects\Sesion4\app\src\main\res\raw`. The "New name" field is set to `musica.mp3`. The bottom status bar indicates "Build: completed successfully at 24/01/2019 10:21".

Android Studio interface showing the project structure and the `ServicioMusica.java` file. The `raw` directory is selected in the Project view. The `ServicioMusica.java` file is open in the editor, showing the `onCreate()` method where a `MediaPlayer` is created from `R.raw.audio`. A "Copy" dialog box is displayed, showing the source file `C:\Users\TOSHIBA\Downloads\musica.mp3` and the destination directory `E:\Users\usuario\AndroidStudioProjects\Sesion4\app\src\main\res\raw`. The "New name" field is set to `musica.mp3`. The bottom status bar indicates "Build: completed successfully at 24/01/2019 10:21".

Servicio.class

```
public class Servicio extends Activity {
    @Override public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_servicio);

        Button arrancar = (Button) findViewById(R.id.boton_arrancar);
        arrancar.setOnClickListener(new View.OnClickListener() {
            public void onClick(View view) {
                startService(new Intent(Servicio.this,
                    ServicioMusica.class));
            }
        });
        Button detener = (Button) findViewById(R.id.boton_detener);
        detener.setOnClickListener(new View.OnClickListener() {
            public void onClick(View view) {
                stopService(new Intent(Servicio.this,
                    ServicioMusica.class));
            }
        });
    }
}
```

activity_servicio.xml

```
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Servicio de reproducción de música"/>
    <Button android:id="@+id/boton_arrancar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Arrancar servicio"/>
    <Button android:id="@+id/boton_detener"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Detener servicio"/>
</LinearLayout>
```

ServicioMusica.class

```
public class ServicioMusica extends Service {
    MediaPlayer reproduktor;
    @Override public void onCreate() {
        Toast.makeText(this, "Servicio creado",
            Toast.LENGTH_SHORT).show();
        reproduktor = MediaPlayer.create(this, R.raw.musica);
    }
    @Override
    public int onStartCommand(Intent intenc, int flags, int idArranque) {
        Toast.makeText(this, "Servicio arrancado " + idArranque,
            Toast.LENGTH_SHORT).show();
        reproduktor.start();
        return START_STICKY;
    }
    @Override public void onDestroy() {
        Toast.makeText(this, "Servicio detenido",
            Toast.LENGTH_SHORT).show();
        reproduktor.stop();
    }
    @Override public IBinder onBind(Intent intencion) {
        return null;
    }
}
```

Añadimos al manifest

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportRtl="true"
    android:theme="@style/AppTheme">
    <service android:name=".ServicioMusica" />
    <activity android:name=".Servicio">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

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



Iniciamos el Servicio
Destruya la actividad
Ingresamos a opciones de desarrollador>Servicios en ejecución

Notificaciones

Para Android 8 en adelante

Definimos antes del Oncreate

```
private static final int ID_NOTIFICACION_CREAR = 1;
public static final String NOTIFICATION_CHANNEL_ID = "1000";
public static final String NOTIFICATION_CHANNEL_NAME = "UNJBG";
```

Añadimos al OnStarCommand

```
@Override
public int onStartCommand(Intent intenc, int flags, int idArraque) {
    NotificationCompat.Builder notific = new NotificationCompat.Builder(this)
        .setContentTitle("Creando Servicio de Música")
        .setSmallIcon(R.mipmap.ic_launcher)
        .setContentText("información adicional");
    NotificationManager notificationManager = (NotificationManager)
        getSystemService(Context.NOTIFICATION_SERVICE);
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        NotificationChannel notificationChannel =
            new NotificationChannel(
                NOTIFICATION_CHANNEL_ID,
                NOTIFICATION_CHANNEL_NAME,
                NotificationManager.IMPORTANCE_LOW);
        notificationChannel.enableLights(true);
        notificationChannel.setLightColor(R.color.colorAccent);
        notificationManager.createNotificationChannel(notificationChannel);
        notific.setChannelId(NOTIFICATION_CHANNEL_ID);
    }
    notificationManager.notify(ID_NOTIFICACION_CREAR, notific.build());
    Toast.makeText(this, "Servicio arrancado " + idArraque,
        Toast.LENGTH_SHORT).show();
    reproductor.start();
    return START_STICKY;
}
```

Notificaciones –

Lanzando una actividad

Eliminando notificación

Añadimos al OnStarCommand

```
@Override
public int onStartCommand(Intent intenc, int flags, int idArraque) {
    NotificationCompat.Builder notific = new NotificationCompat.Builder(this)
        .setContentTitle("Creando Servicio de Música")
        .setSmallIcon(R.mipmap.ic_launcher)
        .setContentText("información adicional");
    // Para lanzar una actividad
    PendingIntent intencionPendiente = PendingIntent.getActivity(
        this, 0, new Intent(this, Servicio.class), 0);
    notific.setContentIntent(intencionPendiente);
}
```

Añadimos al onDestroy→ para eliminar la notificación

```
@Override public void onDestroy() {  
    Toast.makeText(this, "Servicio detenido",  
        Toast.LENGTH_SHORT).show();  
    reproductor.stop();  
    //Eliminando la notificación  
    NotificationManager notificationManager = (NotificationManager)  
        getSystemService(Context.NOTIFICATION_SERVICE);  
    notificationManager.cancel(ID_NOTIFICACION_CREAR);  
}
```

Existen más opciones para la notificación que según la versión de Android pueden no estar disponibles

```
@Override
public int onStartCommand(Intent intenc, int flags, int idArranque) {
    NotificationCompat.Builder notific = new NotificationCompat.Builder(this)
        .setContentTitle("Creando Servicio de Música")
        .setSmallIcon(R.mipmap.ic_launcher)
        .setContentText("información adicional")
        .setLargeIcon(BitmapFactory.decodeResource(getResources(),
            android.R.drawable.ic_media_play))
        .setWhen(System.currentTimeMillis() + 1000 * 60 * 60)
        .setContentInfo("más info")
        .setTicker("Texto en barra de estado")
        .setDefaults(Notification.DEFAULT_SOUND)
        .setDefaults(Notification.DEFAULT_VIBRATE);
}
```

Servicios Android 8.1 superior

Añadimos al manifest

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

Modificamos en el Oncreate el setonclick listener de arrancar

```
arrancar.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View view) {  
  
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {  
            startForegroundService(new Intent(Servicio.this, ServicioMusica.class));  
        } else {  
            startService(new Intent(Servicio.this,  
                                    ServicioMusica.class));  
        }  
    }  
});
```


Añadir antes del **return** *START_STICKY*

```
startForeground(101, notific.build())  
    );
```

Receptor de Anuncios

Ejemplo1

Añadimos al Manifest un permiso y el Receiver

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

```
<receiver android:name="ReceptorLlamadas" >  
  <intent-filter >  
    <action android:name="android.intent.action.PHONE_STATE"/>  
  </intent-filter>  
</receiver>
```

Creamos la clase ReceptorLlamadas

```

public class ReceptorLlamadas extends BroadcastReceiver {
    public static final String NOTIFICATION_CHANNEL_ID = "1000";
    public static final String NOTIFICATION_CHANNEL_NAME = "UNJBG";
    @Override public void onReceive(Context context, Intent intent) {
        // Sacamos información de la intención
        String estado = ""; numero = "";
        Bundle extras = intent.getExtras();
        if (extras != null) {
            estado = extras.getString(TelephonyManager.EXTRA_STATE);
            if (estado.equals(TelephonyManager.EXTRA_STATE_RINGING)) {
                numero = extras.getString(
                    TelephonyManager.EXTRA_INCOMING_NUMBER);
                String info = estado + " " + numero;
                Log.d("ReceptorAnuncio", info + " intent=" + intent);
                // Creamos Notificación
                NotificationCompat.Builder notificacion = new
                    NotificationCompat.Builder(context)
                        .setContentTitle("Llamada entrante ")
                        .setContentText(info)
                        .setSmallIcon(R.mipmap.ic_launcher)
                        .setContentIntent(PendingIntent.getActivity(context, 0,
                            new Intent(context, Servicio.class), 0));
                NotificationManager notificationManager = (NotificationManager)
                    context.getSystemService(Context.NOTIFICATION_SERVICE);
                if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
                    NotificationChannel notificationChannel =
                        new NotificationChannel(
                            NOTIFICATION_CHANNEL_ID,
                            NOTIFICATION_CHANNEL_NAME,
                            NotificationManager.IMPORTANCE_LOW);
                    notificationChannel.enableLights(true);
                    notificationChannel.setLightColor(R.color.colorAccent);
                    notificationManager.createNotificationChannel(notificationChannel);
                    notificacion.setChannelId(NOTIFICATION_CHANNEL_ID);
                }
                notificationManager.notify(1, notificacion.build());
            }
        }
    }
}

```

Para Android 9 o superior

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

Receptor de Anuncios

Ejemplo 2

Añadimos al Manifest un permiso y el Receiver

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

```
<receiver android:name="ReceptorSMS" >  
  <intent-filter>  
    <action android:name="android.provider.Telephony.SMS_RECEIVED"/>  
  </intent-filter>  
</receiver>
```


Creamos la clase ReceptorSMS

```
public class ReceptorSMS extends BroadcastReceiver {  
    @Override public void onReceive(Context context, Intent intent) {  
        Intent i = new Intent(context, Servicio.class);  
        i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);  
        context.startActivity(i);  
    }  
}
```

Receptor de Anuncios Leyendo el SMS Ejemplo 2.1

Añadimos al final del onReceive de ReceptorSMS

```
final Bundle bundle = intent.getExtras();
try {
    if (bundle != null) {
        final Object[] pduObj = (Object[]) bundle.get("pdu");
        for (int j = 0; j < pduObj.length; j++) {
            SmsMessage currentMessage = SmsMessage.createFromPdu((byte[]) pduObj[j]);
            String phoneNumber = currentMessage.getDisplayOriginatingAddress();
            String senderNum = phoneNumber;
            String message = currentMessage.getDisplayMessageBody();
            Log.i("receptorsms", "senderNum: " + senderNum + "; message: " + message);
        }
    }
} catch (Exception e) {
    Log.e("SmsReceiver", "Exception smsReceiver" + e);
}
```

Activando pantalla aun bloqueada

Añadir al Oncreate de la Actividad Servicio

```
@Override public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    getWindow().addFlags(  
        WindowManager.LayoutParams.FLAG_TURN_SCREEN_ON | WindowManager.LayoutParams.FLAG_DISMISS_KEYGUARD |  
        WindowManager.LayoutParams.FLAG_SHOW_WHEN_LOCKED);  
    setContentView(R.layout.activity_servicio);  
}
```



Universidad Nacional
Jorge Basadre Grohmann



Gracias por su atención.

MSc. Ing. Wilson César Callisaya Choquecota

nosliwsys@gmail.com

952000243 – Tacna Perú