

Crear un servicio en Android.

Un Servicio en Android.

Un servicio es un componente de una aplicación el cual se ejecuta en segundo plano (background), este puede realizar las mismas acciones que un Activity con la única diferencia que este no proporciona ningún tipo de interfaz de usuario.

Los servicios son útiles para acciones que se requieran realizar durante un tiempo en background, sin tener en cuenta lo que está en pantalla tales como:

Ejecutar transacciones de red, reproducir música, ejecutar archivos, ejecutar procesos, todo desde background.

Un servicio es ejecutado desde el hilo principal, este no crea un hilo propio y no corre en un proceso separado, esto significa que si un servicio realiza un trabajo intensivo en el CPU es necesario crear un nuevo Hilo dentro del servicio para poder reducir el riesgo de un ANR (Application Not Responding) en el dispositivo.

Ejemplo Practico

Para poder crear un nuevo servicio es necesario que en nuestro proyecto agreguemos una clase más en la cual implementaremos nuestro servicio, puede estar en el paquete principal u otro paquete o sub-paquete (No importa donde se encuentre).

En este caso llamaré a esta clase como ServiceDemo el cual hereda de Services.

Se importan las librerías correspondientes y agregamos el método onBind el cual recibe como parámetro un objeto Intent y regresará un objeto IBinder.

```
import android.app.Service;
import android.content.Intent;
import android.os.IBinder;

public class ServiceDemo extends Service{

    public IBinder onBind(Intent intent) {
        return null;
    }
}
```

El método OnBind es un método abstracto el cual nos regresa el canal de comunicación del servicio y sirve para establecer comunicación entre servicios (para esto es necesario definir un AIDL). Colocamos null para declarar que en esta ocasión no estableceremos esa comunicación.

más detalles:

Creamos los siguientes métodos dentro de nuestra clase:

```
public void onCreate(){
    super.onCreate();
}

public void onStart(Intent intent, int startId){
    System.out.println("El servicio a Comenzado");
}
```

```
        this.stopSelf();  
    }  
    public void onDestroy() {  
        super.onDestroy();  
        System.out.println("El servicio a Terminado");  
    }  
}
```

Estos métodos nos permitirán crear el servicio, ejecutarlo y detenerlo, colocamos unas líneas de código más para poder ver el comportamiento de la aplicación (Unos mensajes).

La llamada del método **stopSelf** dentro del método **onStart** nos permite detener el servicio dentro del mismo.

Para poder utilizar nuestro servicio necesitaremos colocar la siguiente línea, en mi caso en el MainActivity.

```
startService(new Intent(this, ServiceDemo.class));
```

El método **startService** permite poner en marcha un servicio, en caso no deseemos detener el servicio en el mismo, siempre podemos utilizar el método **stopService** que recibe como parámetro un objeto Intent.

```
stopService(new Intent(this, ServiceDemo.class));
```

Sin embargo hasta este momento no es posible ver los mensajes por parte de nuestro servicio, debido que es necesario registrarlo en el Manifest con la siguiente instrucción:

```
<service android:name=".ServiceDemo"></service>
```

Coloco el .(punto) para indicar que la clase que usaré como Servicio se encuentra en el paquete principal, en caso la clase se ubique en otro paquete, la sentencia quedaría de la siguiente manera:

```
<service android:name="otroPaquete.ServiceDemo"></service>
```

Con ayuda de **LogCat** podremos visualizar los mensajes obtenidos en esta prueba.

Como recordatorio es importante mencionar que un servicio no está diseñado para realizar tareas con costos computaciones altos ya que esto ocasionará problemas en la aplicación ya que como se mencionó el servicio corre en el Hilo principal de la aplicación, para solucionar esto es posible crear un nuevo Hilo dentro del servicio o Utilizar la clase **IntentService** la cual crea un servicio en un nuevo hilo.