Service

Bruno Lopes Alcantara Batista

Prazer em conhecer,

Bruno Lopes A Batista

Me. Ciência da Computação - UECE

Esp. em Desenvolvimento Web com JavaEE - FJN

Bel. em Sistemas de Informação - FJN

Professor Pesquisador - LARCES

Gerente de Projetos - LARCES

Gerente de Projetos - Athomustec

bruno@larces.uece.br

http://www.larces.uece.br/bruno

Copyleft

Agenda

- Introdução
- Criando um Service
  - Criando um Started Service
  - Criando um Bound Service
- Let's try it

Service

Introdução

Service

Introdução

- É uma aplicação que executa uma operação longa em background.
- Não fornece uma interface de usuário.
- Um outro componente da aplicação pode iniciar um Service.
- Adicionalmente um componente pode vincular-se a um Service para realizar alguma interação:
  - Manipular conexões de rede;
  - Executar músicas;
  - E/S de arquivos;
  - Etc

Service

Introdução

- Um Service pode essencialmente ter duas formas:
  - **Started:** Iniciado quando um componente o inicia executando o método *startService().*
  - **Bound:** Iniciado quando um componente o inicia executando o método *bindService()*.
- O Started Service geralmente utilizado quando o service não retorna nenhum resultado.
- O Bound service é geralmente utilizado quando o service retorna algum resultado para quem se vinculou ao mesmo.

Service

Criando um Service

Service

Criando um Service

- Para criar um service, é necessário criar uma classe Java que extende da classe Service.
- Nessa classe, deve-se sobrescrever alguns métodos de callback.
- Esses métodos manipulam alguns aspectos chaves do ciclo de vida do Service.
- Além de prover um mecanismo para os componentes se vincularem ao Service, se necessário.

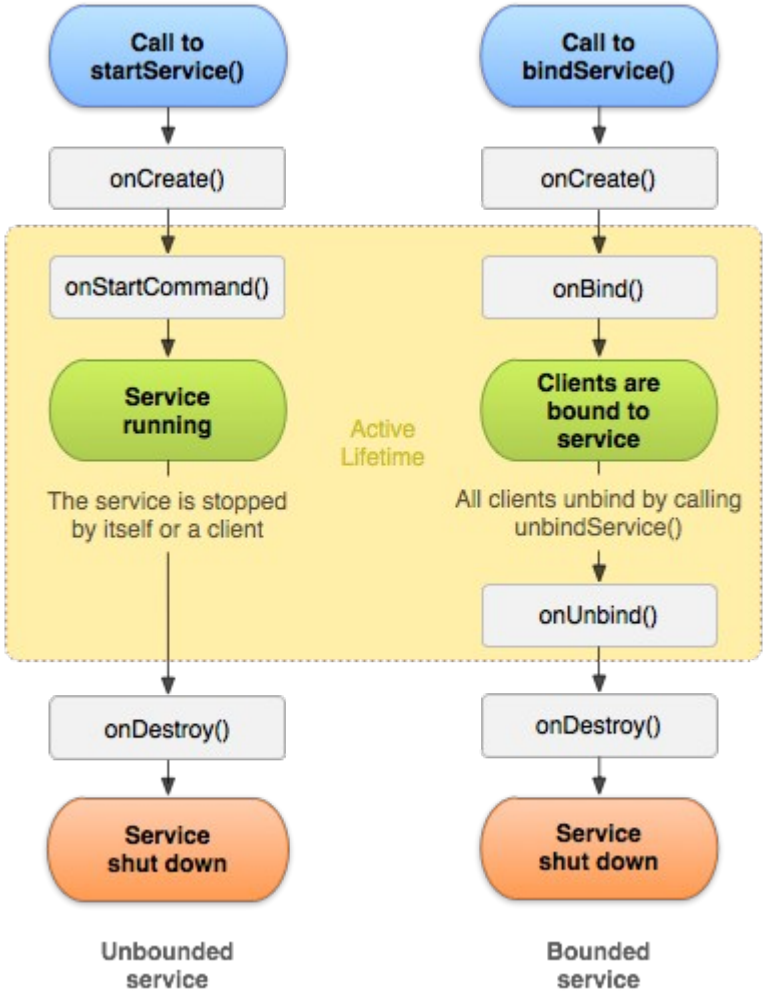
Service

Criando um Service

- Os métodos de callcack mais importantes são:
  - **onStartCommand():** É chamado quando outro componente requisita que o Service Seja iniciado. O programador deve implementar a condição de parada do Service, executando o método **stopService()** ou **stopSelf()**
  - **onBind():** É chamado quando um outro componente deseja se vincular ao Service.
  - **onCreate():** É chamado quando o Service é criado pela primeira vez.
  - **onDestroy():** É chamado quando o Service não é mais necessário.

Service

Criando um Service



Service

Criando um Service

- Assim como a Activity, um service deve ser declarado no AndroidManifest.xml

```
<manifest ... >
  ...
  <application ... >
      <service android:name=".ExampleService" />
      ...
  </application>
</manifest>
```

Service

Criando um Started Service

Service

Criando um Started Service

- Criar uma classe que extende de service e implementar o método onStartCommand():

```
11  public class StartedService extends Service {
12
13
14      @Override
15      public int onStartCommand(Intent intent, int flags, int startId) {
16
17          new Thread(new Runnable() {
18
19              @Override
20              public void run() {
21                  for (int i = 0; i < 100; i++) {
22                      try {
23                          Log.i("APP", "Value: " + i);
24                          Thread.sleep(1000);
25                      } catch (InterruptedException e) {
26                          e.printStackTrace();
27                      }
28                  }
29              }
30
31          }).start();
32
33          return Service.START_NOT_STICKY;
34      }
35
36      @Override
37      public IBinder onBind(Intent intent) {
38          return null;
39      }
40  }
```

Service

Criando um Started Service

- Após declarar o Service no arquivo de manifesto, basta inicia-lo:

```
 8  public class SplashScreen extends Activity {
 9
10      @Override
11      protected void onCreate(Bundle savedInstanceState) {
12
13          super.onCreate(savedInstanceState);
14          requestWindowFeature(Window.FEATURE_NO_TITLE);
15          setContentView(R.layout.activity_splash_screen);
16
17          Intent it = new Intent(this, StartedService.class);
18          startService(it);
19
20      }
21
22  }
23
```

Service

Criando um Bound Service

Service

Criando um Started Service

- Criar uma classe que extende de service e implementar o método onBind():

```
10  public class BoundService extends Service {
11
12      private final IBinder binder;
13      private boolean stop;
14
15      public BoundService() {
16          this.binder = new LocalBinder();
17          this.stop = false;
18      }
19
20      @Override
21      public IBinder onBind(Intent intent) {
22          return binder;
23      }
24
25
26      public class LocalBinder extends Binder{
27          public BoundService getService(){
28              return BoundService.this;
29          }
30      }
```

```
31
32      public void startCount(){
33          int i = 0;
34          while(!stop){
35              try {
36                  Thread.sleep(1000);
37                  Log.i("APP", "Value: " + i);
38                  i++;
39              } catch (InterruptedException e) {
40                  e.printStackTrace();
41              }
42          }
43      };
44
45      public void stop(){
46          this.stop = true;
47      }
48
49  }
50
```

Service

Criando um Started Service

- Após declarar o Service no arquivo de manifesto, basta inicia-lo:

```
14  public class SplashScreen extends Activity {
15
16      private BoundService mService;
17      private boolean mBound;
18      private boolean mCounterStarted;
19
20      @Override
21      protected void onCreate(Bundle savedInstanceState) {
22
23          super.onCreate(savedInstanceState);
24          requestWindowFeature(Window.FEATURE_NO_TITLE);
25          setContentView(R.layout.activity_splash_screen);
26          mCounterStarted = false;
27
28          findViewById(R.id.button).setOnClickListener(new View.OnClickListener() {
29              @Override
30              public void onClick(View v) {
31                  if(mBound){
32                      if(!mCounterStarted){
33                          mService.startCount();
34                          mCounterStarted = true;
35                      } else {
36                          mService.stopCount();
37                          mCounterStarted = false;
38                      }
39                  }
40              }
41          });
42
43      }
```

Service

Criando um Started Service

- Após declarar o Service no arquivo de manifesto, basta inicia-lo (continuação):

```
44
45      @Override
46      protected void onStart() {
47          super.onStart();
48          Intent it = new Intent(this, BoundService.class);
49          bindService(it, mConnection, Context.BIND_AUTO_CREATE);
50      }
51
52      @Override
53      protected void onStop() {
54          super.onStop();
55          if(mBound){
56              unbindService(mConnection);
57          }
58      }
59
60      private ServiceConnection mConnection = new ServiceConnection() {
61          @Override
62          public void onServiceConnected(ComponentName name, IBinder service) {
63              BoundService.LocalBinder binder = (BoundService.LocalBinder) service;
64              mService = binder.getService();
65              mBound = true;
66          }
67
68          @Override
69          public void onServiceDisconnected(ComponentName name) {
70              mBound = false;
71          }
72      };
73
74  }
```

Service

Let's try it!

Service

Let's try it

- Implementar um exemplo simples do Started Service;
- Implementar um exemplo simples do Bound Service;
- Implementar o Bound Service na aplicação **Droid Weather**

Obrigado!


danke
謝謝
ngiyabonga
teşekkür ederim
спасибо
dank je
thank you
gracias
tapadh leat
bedankt
hvala
mauruuru
mochchakkeram
dziękuję
go raibh maith agat
sagolun
sukriya
kop khun krap
arigatô
takk
dakujem
obrigado
grazie
мерси
terima kasih
감사합니다
ευχαριστώ
merci