

Android Services

Marcio E. F. Maia
marcioefmaia@ufc.br

Introdução

Service

- Executa operações mais longas
 - Parte do código executada em *background*
- Código que pode ser executado por outras aplicações
- Pode ter ciclo de vida diferente da aplicação que a iniciou
 - Mesmo que aplicação seja finalizada, Service continua executando
- Não possui interface gráfica

Introdução

Service

- Parte da aplicação que monitora determinado ação do usuário
 - Entrou em uma área delimitada
 - Algum evento marcado no calendário
 - Alguém adicionou o usuário no facebook
 - Implementar a leitura a um arquivo, transação pela rede ou tocar uma música
- Qualquer coisa que você deseja que permaneça sendo executado quando aplicação é fechada



Introdução

Service

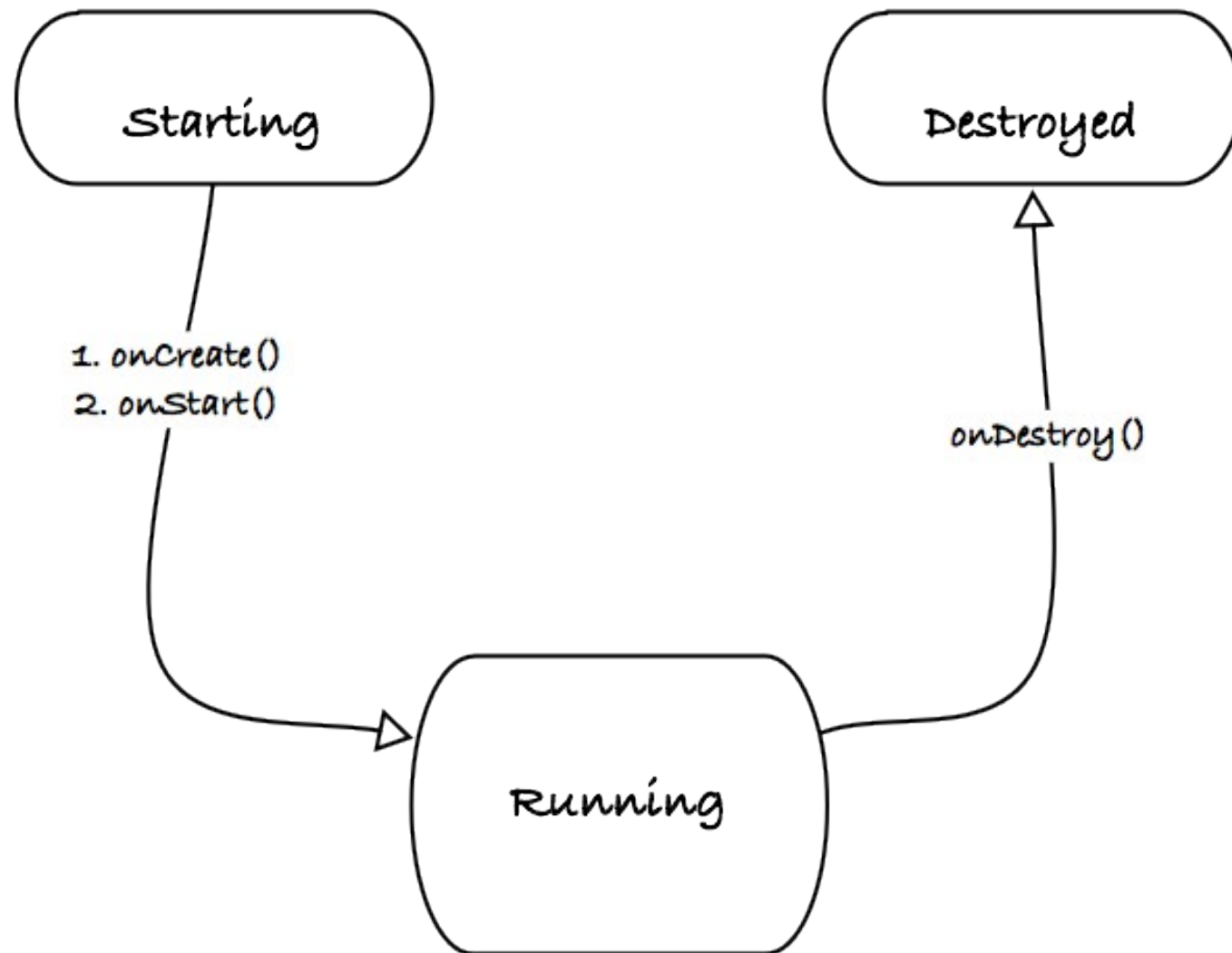
- Services rodam por um tempo maior do que a aplicação
 - ... mas use com precaução
 - Eles consomem recursos e bateria
- Usuário pode não saber o que está acontecendo
 - Dispositivo pode ficar lento e bateria acabar muito rápido
 - VM pode matar services consumindo muitos recursos, mas apenas em situações extremas



Ciclo de Vida

- Services são independentes da aplicação
- Podem rodar em um processo separado
 - Mesmo que aplicação seja terminada, service pode continuar rodando
- Possuem semelhança com as Threads
 - Rodam em background
 - Independente da aplicação
- **Principal diferença é que os Services são independentes**

Ciclo de vida



Criando um Service

- Estender a classe Service

```
package br.ufc.quixada.androidservices.services;  
  
import android.app.Service;  
  
public class AndroidServices extends Service {  
  
}
```

- Declarar o service no AndroidManifest.xml

```
<service android:name=".services.AndroidServices"></service>
```

Inicializando um Service

- Inicializar um Service

```
public void onClickStartService( View view ){  
    Intent startServiceIntent = new Intent( this, AndroidServices.class );  
    startService( startServiceIntent );  
}
```

- Esse código chama os métodos onCreate e onStartCommand do Service
 - onCreate() é executado apenas uma vez enquanto o Service estiver rodando
 - onStartCommand() é executando toda vez que método startService for chamado

Inicializando um Service

- onCreate()
 - Colocar código para alocar recursos
 - Como abrir conexão com o banco de dados ou servidor
- onStartCommand()
 - Inicializar thread que faz o comportamento esperado
 - Deve tratar se já foi executando anteriormente, pois esse método é executado sempre alguém chama o método startService(intent)

Parando um Service

- Finalizar o Service

```
public void onClickStopService( View view ){  
    Intent stopServiceIntent = new Intent( this, AndroidServices.class );  
    stopService( stopServiceIntent );  
}
```

- Esse código chama o método onDestroy() do Service
 - Desalocar recursos alocados no método onCreate()