



Desenvolvimento de Software para Dispositivos Móveis

Aula 5 - Android - Activities e Intents



Professor: Anderson Almada

Introdução

- A classe Activity é um componente crucial de um aplicativo Android
- As atividades são lançadas. Gerencia a interface com o usuário. Recebe as requisições, as trata e processa
- O sistema Android inicia o código em uma instância Activity invocando métodos de retorno de chamada específicos que correspondem a estágios específicos de seu ciclo de vida

Ciclo de vida

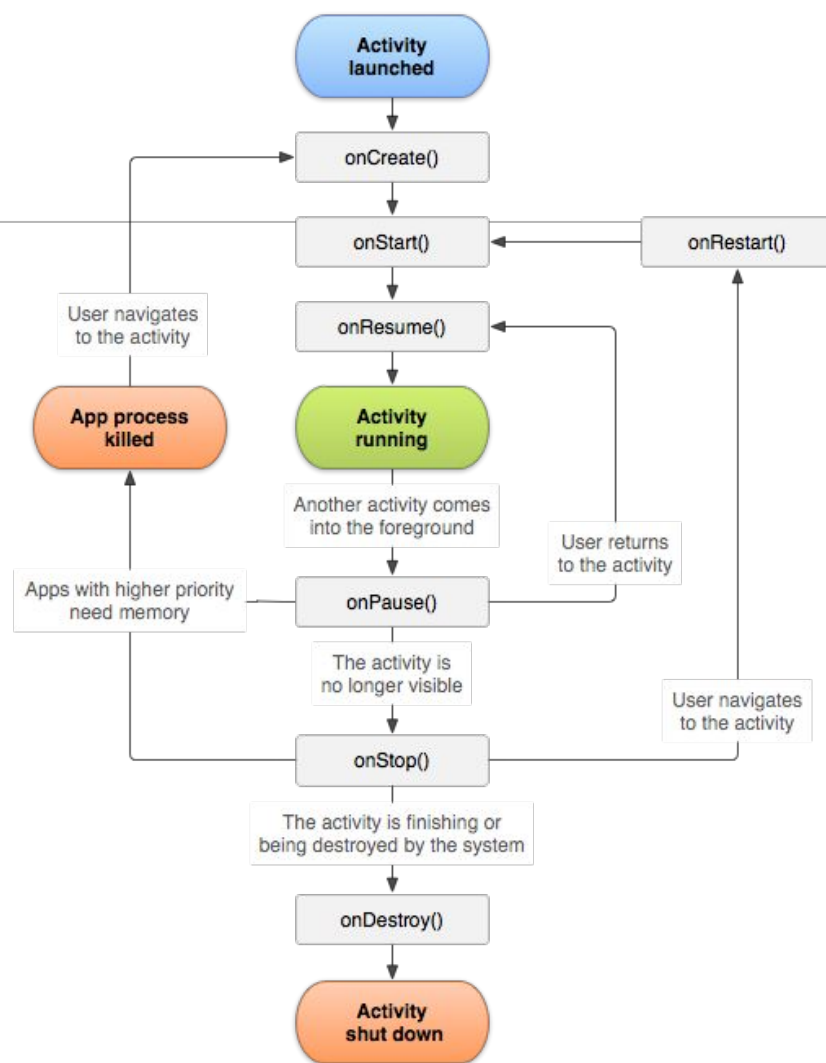
- **onCreate():** É a primeira função a ser executada. Responsável por carregar os layouts XML e outras operações de inicialização. Executada apenas uma vez.
- **onStart():** É chamada imediatamente após a onCreate() – e também quando uma Activity que estava em background volta a ter foco.
- **onResume():** Assim como a onStart(), é chamada na inicialização da Activity. A onStart() só é chamada quando a Activity não estava mais visível e volta a ter o foco, a onResume() é chamada nas “retomadas de foco”.

Ciclo de vida

- **onPause():** É a primeira função a ser invocada quando a Activity perde o foco (isso ocorre quando uma nova Activity é iniciada).
- **onStop():** Só é chamada quando a Activity fica completamente encoberta por outra Activity.
- **onDestroy():** A última função a ser executada. Depois dela, a Activity é considerada “morta” – ou seja, não pode mais ser relançada.

Ciclo de vida

- **onRestart()**: Chamada imediatamente antes da `onStart()`, quando uma Activity volta a ter o foco depois de estar em background.



Logcat

- O Logcat é uma ferramenta de linha de comando que gera um registro de mensagens do sistema, inclusive rastros de pilha quando o dispositivo apresenta um erro e mensagens escritas pelo seu aplicativo com a classe Log.

Logcat

- **Verbose** - Exibe todas as mensagens de registro (padrão).
- **Debug** - Exibe as mensagens de registro da depuração úteis apenas durante o desenvolvimento, além de mensagens de nível inferior dessa lista.
- **Info** - Exibe mensagens de registro esperadas para uso regular
- **Warn** - Exibe possíveis problemas que ainda não são erros
- **Error** - Exibe problemas que causaram erros
- **Assert** - Exibe problemas que o desenvolvedor acredita que nunca acontecerão.

Logcat

- **Log.e**(String, String) (erro)
- **Log.w**(String, String) (aviso)
- **Log.i**(String, String) (informação)
- **Log.d**(String, String) (depuração)
- **Log.v**(String, String) (detalhado)

Intent

- Intent é a mensagem transmitida entre componentes, como atividades, provedores de conteúdo, receptores de broadcast, serviços, etc.
- Geralmente é usada com o método `startActivity ()` para invocar atividade
- O significado de intenção do dicionário é intenção ou propósito. Então, isso pode ser descrito como a intenção de fazer ação.

Intent

- As intenções do Android são usadas principalmente para:
 - Começar um serviço
 - Lançar uma atividade
 - Exibir uma página da web
 - Exibir uma lista de contatos
 - Transmitir uma mensagem
 - Disque um telefonema etc.

Intent

- Dois tipos

- Implícita

```
Intent intent=new Intent(Intent.ACTION_VIEW);  
intent.setData(Uri.parse("http://www.javatpoint.com"));  
startActivity(intent);
```

- Explícita

```
Intent i = new Intent(getApplicationContext(), ActivityTwo.class);  
startActivity(i);
```

Exercício 1

- Crie um novo projeto
 - PrimeiroApp
 - package: br.ufc.crateus.primeiroapp
- No layout
 - Crie uma caixa de texto (EditText)
 - Crie um botão com texto **OPEN**

Exercício 1

- No java
 - Recupere os dois elementos
 - Insira o evento de escuta de clique no botão
 - Faça com que a aplicação tenha a intenção de visualizar no navegador o endereço digitado na caixa de texto

Exercício 2

- Crie um novo projeto
 - SegundoApp
 - package: br.ufc.crateus.segundoapp
- No layout
 - Crie duas caixas de texto (EditText)
 - Crie um botão com texto **SEND**

Exercício 2

- No java
 - Recupere os três elementos
 - Insira o evento de escuta de clique no botão
 - Faça com que a aplicação tenha a intenção de iniciar uma nova activity (SecondActivity) passando como parâmetro os dois conteúdos das caixas de texto

Exercício 2

- No java
 - Se os dois conteúdos forem Strings, SecondActivity deve mostrar o resultado da concatenação entre os dois
 - Se os dois conteúdos forem números, SecondActivity deve mostrar o resultado da soma entre os dois
 - Qualquer outro caso, SecondActivity deve mostrar um Toast informando **“Erro”**

startActivityResult

- Obter resultado de outra atividade. Podemos enviar informações de uma atividade para outra e vice-versa.
- Método android requer um resultado da segunda atividade (atividade a ser invocada).
- Nesse caso, precisamos substituir o método onActivityResult que é chamado automaticamente quando a segunda atividade retorna o resultado.

startActivityForResult

- Primeira Activity (layout)
 - <https://pastebin.com/NCk6syre>
- Segunda Activity (layout)
 - <https://pastebin.com/LkheMNbA>

startActivityForResult

- Primeira Activity (java)
 - <https://pastebin.com/0VHdvdxH>
- Segunda Activity (java)
 - <https://pastebin.com/W2rcKhNX>

startActivityForResult

- Câmera

- Manifest

- ```
<uses-feature android:name="android.hardware.camera" android:required="true" />
```

- <https://pastebin.com/yzRPQMk>

# Link importante

---

- <https://www.javatpoint.com/android-tutorial>
- <https://developer.android.com/guide/components/activities/intro-activities?hl=pt-BR>
- <https://developer.android.com/guide/components/intents-filters?hl=pt-BR>
- <https://stackoverflow.com/questions/1102891/how-to-check-if-a-string-is-numeric-in-java>



# Dúvidas??

E-mail: [almada@crateus.ufc.br](mailto:almada@crateus.ufc.br)