



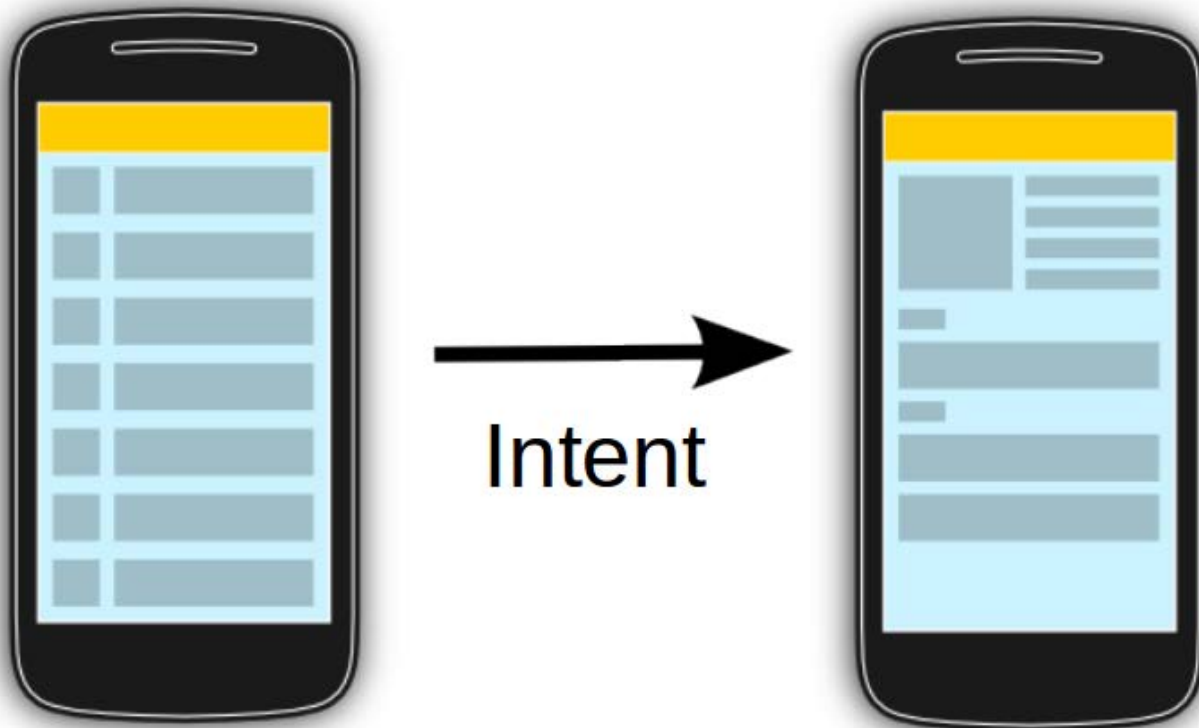
PROGRAMAÇÃO PARA  
DISPOSITIVOS MÓVEIS

# INTENTS

# O que são Intents?

- Atuam como links entre as atividades no Android
- Informam o destino e podem carregar uma pequena quantidade de informação

# Intents



# Que tipos de ações podem iniciar intents?

- Intents podem ser disparados de atividades e de serviços
- O método mais comum é composto por eventos disparados pela atividade em andamento
  - Exemplos?

# Exemplos

- Abrir uma atividade de detalhe;
- Abrir a atividade de configurações da aplicação;
- Abrir uma URL;
- Abrir uma foto;

# Utilizando Intents

- Para utilizar um intent, é necessário instanciar um elemento da classe “Intent”
  - Pacote “android.content.Intent”
- Existem diferentes tipos de intents, específicos para alguns tipos de atividades. Vamos conhecê-los a seguir.

# TIPOS DE INTENTS



# Tipos de Intents

- Intents Explícitos
  - Indicam intents que especificam EXATAMENTE que elemento ou atividade será iniciado. Não há escolha ou mais de uma ação possível neste caso;
  - **Muito utilizados para iniciar uma nova atividade específica (normalmente da própria aplicação).**

# Tipos de Intents

- Intents Implícitos
  - Correspondem a intents em que mais de um elemento pode realizar determinada ação;
  - Comuns em situações onde existe uma ação específica (por exemplo, abrir um link), mas vários elementos podem realiza-la (diversos navegadores, por exemplo)

# Construtores de Intents

- É possível utilizar intents explícitos e implícitos apenas diferenciando-os pelo tipo de construtor utilizado para obter uma instância de um intent, que, em seguida, pode ser utilizada.

# Construtor de Intent Explícito

- Passar o objeto que representa a atividade que está chamando o intent (geralmente, utiliza-se “this” ou “ClasseDaAtividade.this” para indicar a atividade corrente) e a **classe da atividade a iniciar**.
- Esse tipo de chamada corresponde a um **intent “explícito”**

# Construtor de Intent Implícito

- Passar uma **String** (chamada de “ação”), que pode ser configurada no `AndroidManifest.xml`;
- Um elemento que atenda a ação será acionado. Caso mais de um elemento atenda a ação, o usuário poderá escolher qual utilizar;
- Corresponde a um **intent “implícito”**

# Exemplo de Intent Explícito

- Exemplo de intent passando quem chama e a classe destino (explícito):

```
Intent intent = new Intent(MainActivity.this,  
    MinhaAtividade.class);
```

- Neste caso, basta haver tal atividade declarada no AndroidManifest.xml (no Android Studio as atividades são declaradas neste arquivo automaticamente):

```
<activity android:name="tsi.senac.gradehoraria.MinhaAtividade" >  
</activity>
```

# Exemplo de Intent Implícito

- Exemplo de intent passando uma ação (impl.):

```
Intent intent = new
```

```
Intent("tsi.senac.action.MINHA_ATIVIDADE");
```

- No AndroidManifest.xml, a atividade deve ser configurada com um filtro para essa ação:

```
<activity android:name="tsi.senac.gradehoraria.MinhaAtividade" >  
  <intent-filter>  
    <action android:name="tsi.senac.action.MINHA_ATIVIDADE" />  
    <category android:name="android.intent.category.DEFAULT" />  
  </intent-filter>  
</activity>
```

# INICIANDO ATIVIDADES COM INTENTS



# Iniciando uma Atividade

- Primeiro, é necessário obter uma instância do intent para o elemento desejado (implícito ou explícito, conforme visto anteriormente)
- Depois de criado o intent, basta chamar o método “startActivity()” passando o intent criado:

```
Intent intent = new Intent(MainActivity.this, MinhaAtividade.class);  
startActivity(intent);
```

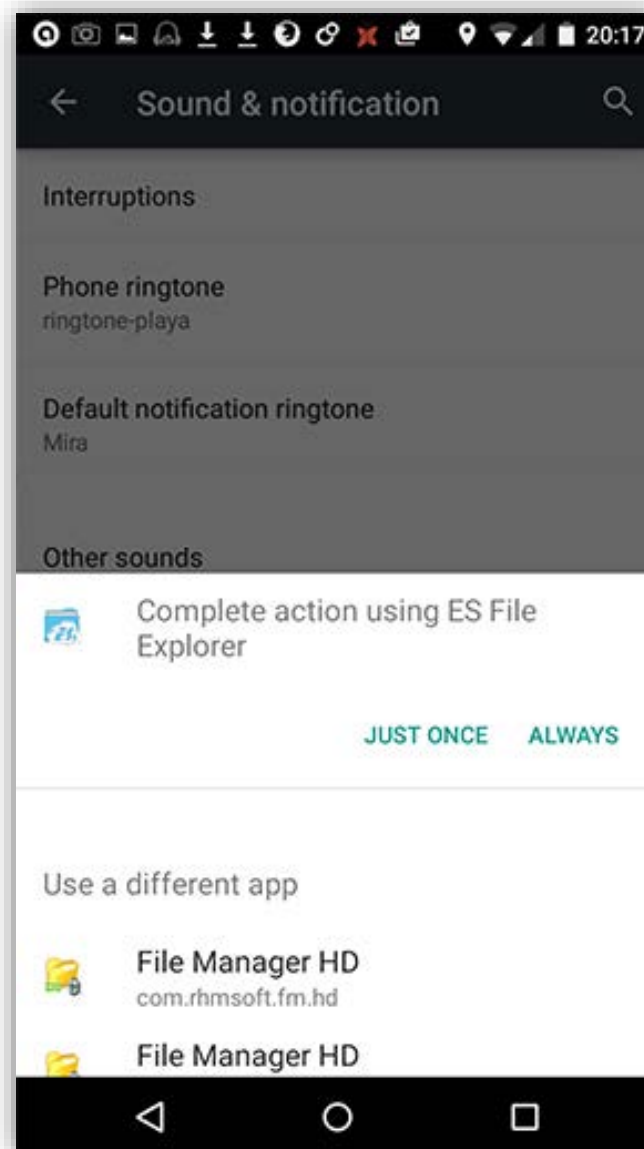
# Executando ações com intents

- Exemplo de intent com mais de uma resposta:

```
Intent i = new Intent(Intent.ACTION_VIEW,  
Uri.parse("http://www.google.com"));
```

- Nesse caso, todas as atividades que sabem tratar URLs (navegadores) irão responder.
  - Caso haja apenas um navegador, o mesmo será aberto imediatamente. Caso contrário, o usuário deverá escolher em uma lista qual elemento abrir

# Diálogo de Seleção de Atividades



# ENVIANDO DADOS COM INTENTS

# Enviando Dados no Intent

- É possível enviar, junto ao intent, uma pequena quantidade de informação para a atividade a ser aberta
- Através do método “putExtra()”, que recebe um nome e o dado que se quer passar:

```
Intent i = new Intent();  
i.putExtra("nome", objetoComODado);  
i.putExtra("id", 5);  
startActivity(intent);
```

# Recebendo Dados do Intent

- Na atividade que foi iniciada com o intent, pode-se recuperar o intent e o dado
- O método depende do tipo de dado passado
- Exemplo com int e String:

```
Intent intent = getIntent();
```

```
int id = intent.getIntExtra("id", 0);
```

```
String name = intent.getStringExtra("name");
```

RECEBENDO DADOS DE UMA ATIVIDADE  
INICIADA COM UM INTENT (RESPOSTA)

# Recebendo Resposta de Atividades

- Se você for esperar um resultado de uma atividade chamada por um intent, é possível utilizar o método “startActivityResult()” (ao invés de “startActivity”, passando o intent e um código único para identificar depois quem está respondendo:

```
startActivityResult(intent, 1);
```



# Passando Dados no Intent

- A atividade iniciada pode retornar o resultado através de um intent também:

```
Intent returnIntent = new Intent();  
returnIntent.putExtra("retorno", "valor de  
retorno");  
setResult(RESULT_OK, returnIntent);
```

# Passando Dados no Intent

- A atividade que chamou recebe o resultado no método call-back de atividade “onActivityResult()”:

```
@Override
```

```
protected void onActivityResult(int requestCode, int resultCode,  
    Intent data) {
```

```
    // Verifica de quem veio a resposta
```

```
    if (requestCode == 1) {
```

```
        // Se foi resposta de sucesso
```

```
        if (resultCode == RESULT_OK) {
```

```
            // Faz alguma coisa
```

```
        }
```

```
    }
```

```
}
```

*"That's all Folks!"*