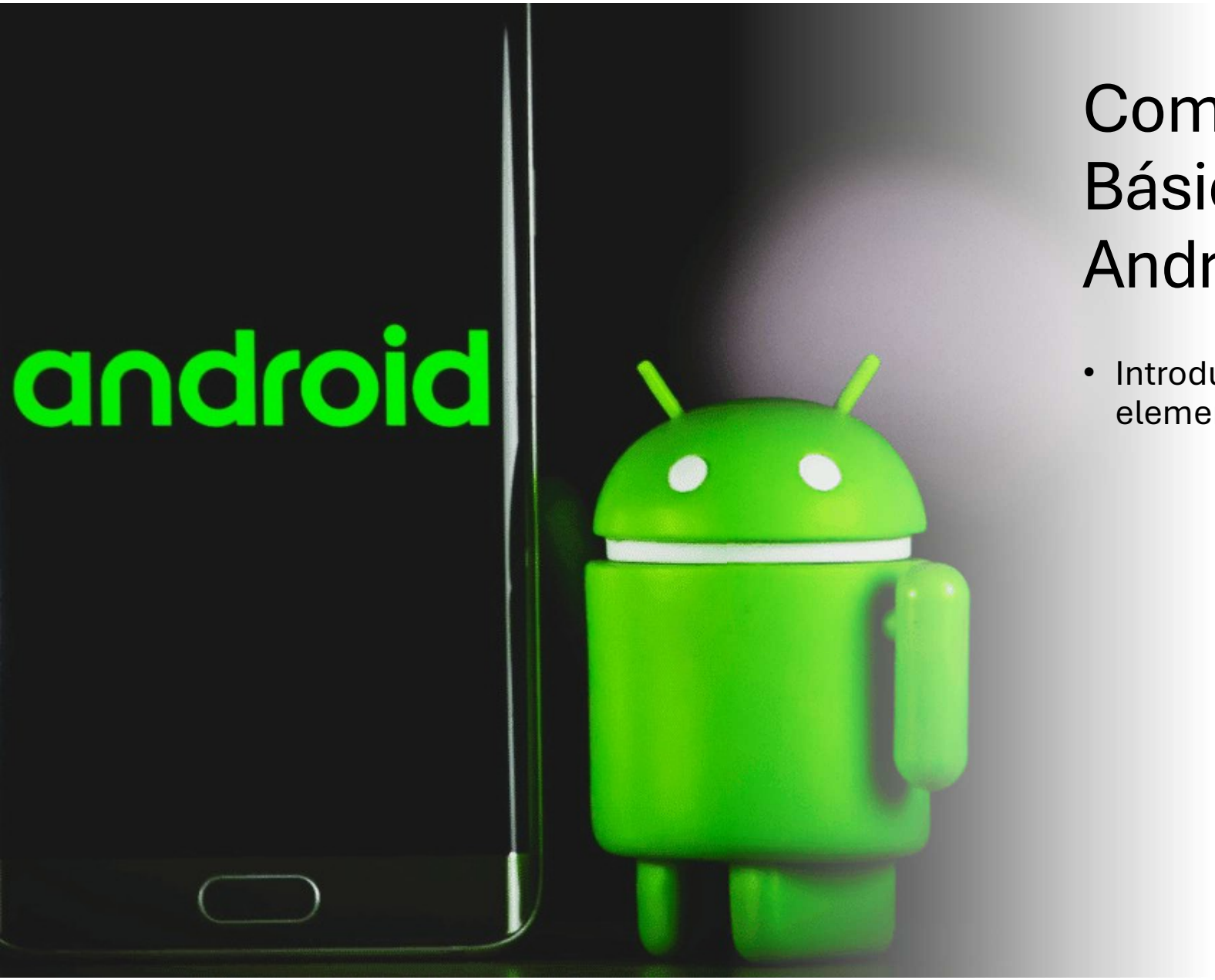
Desenvolvimento de Apps para Mobile

Fabrício Tonetto Londero

Aula 03 - Componentes



Componentes Básicos do Android

- Introdução aos principais elementos de interface gráfica

Objetivos

Compreender os principais componentes visuais do Android

Criar interfaces utilizando XML e Java

Construir um app básico na prática

O que é a Interface do Usuário (UI)?

- É a parte visual do aplicativo
- Conjunto de componentes que permitem interação com o usuário
- Criada com XML (design) e Java/Kotlin (lógica)
- **Exemplos de componentes:**
  - Botões, caixas de texto, imagens, listas, entre outros

TextView: Exibição de texto

Componente Essencial

O TextView é fundamental para a exibição de texto em aplicações Android, servindo como a interface principal para o usuário.

Formatação de Texto

Permite a formatação de texto de várias maneiras, incluindo tamanho, estilo e cor, para melhor apresentação.

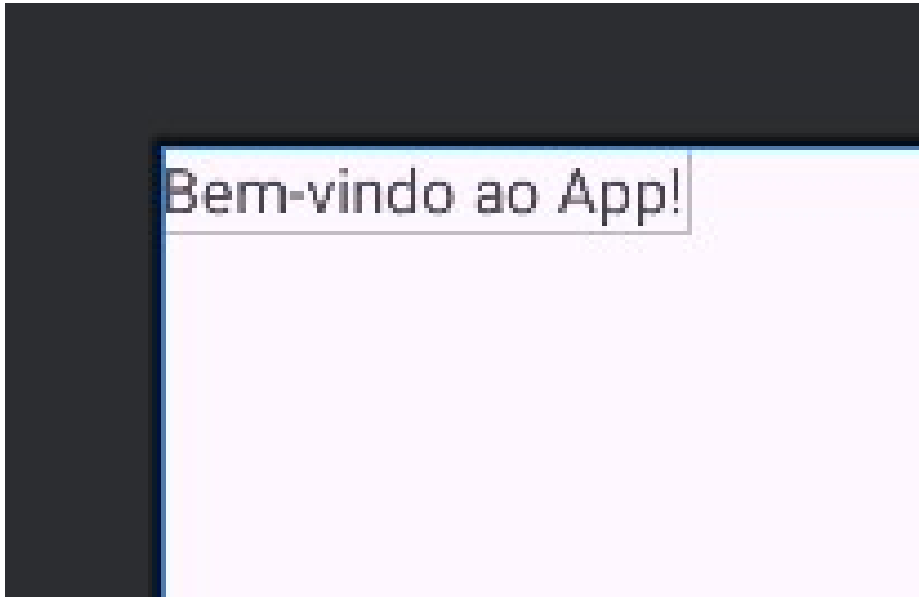
Mostrar Informações

Utilizado para exibir informações e instruções, ajudando os usuários a navegar na aplicação de forma eficaz.

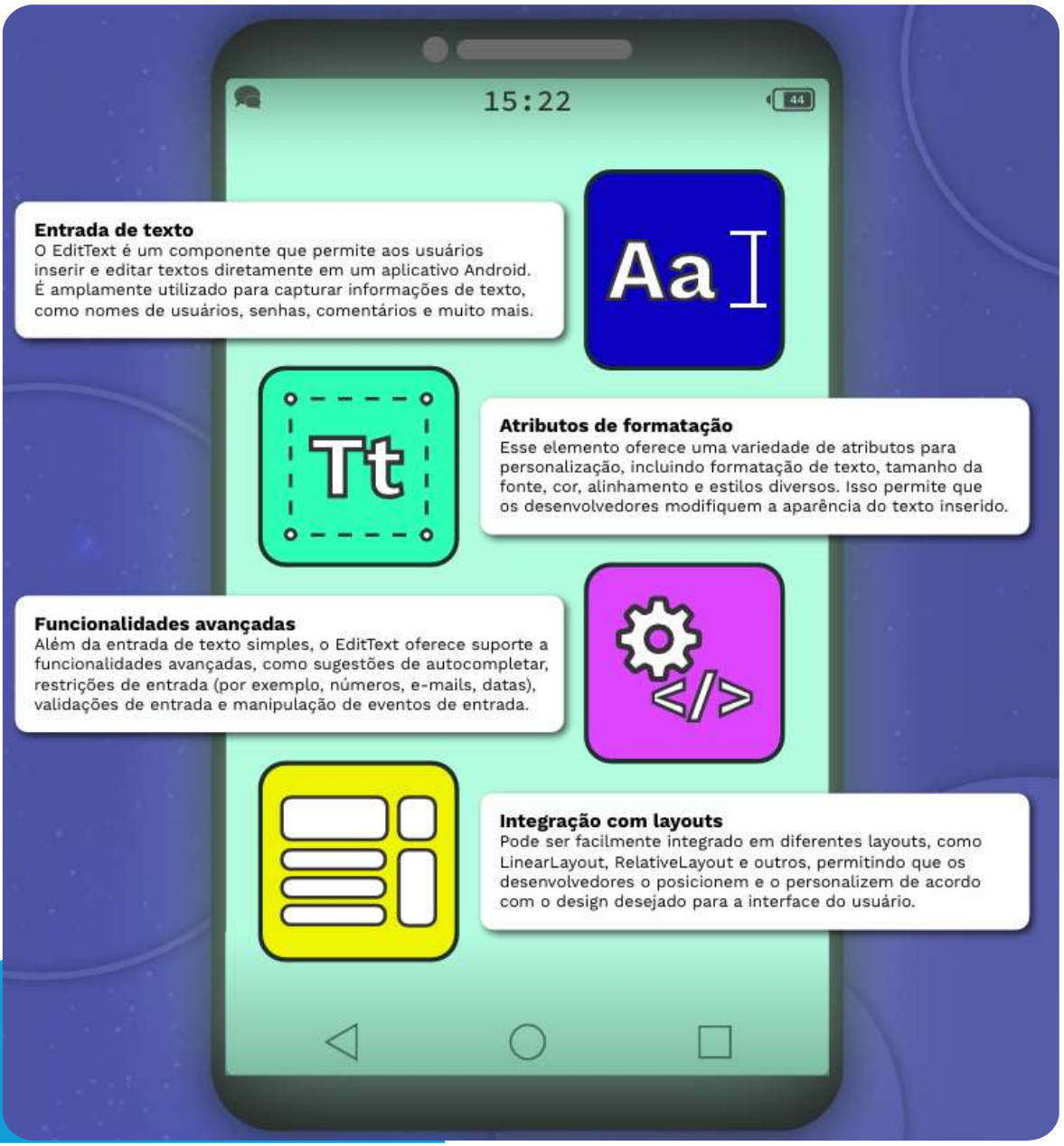
TextView

```
<TextView
android:id="@+id/textViewMensagem"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Bem-vindo ao App!" />
```

```
TextView textViewMensagem = findViewById(R.id.textViewMensagem);
textViewMensagem.setText("Novo texto exibido!");
```



TextView



EditText

Componente EditText

O EditText é uma ferramenta que permite a entrada de texto pelo usuário em aplicativos.

Captura de Dados do Usuário

Utilizado para capturar informações como nome e senha de forma eficiente e segura.

Validação de Entradas

Validar as entradas do usuário é crucial para garantir que os dados sejam corretos e seguros.

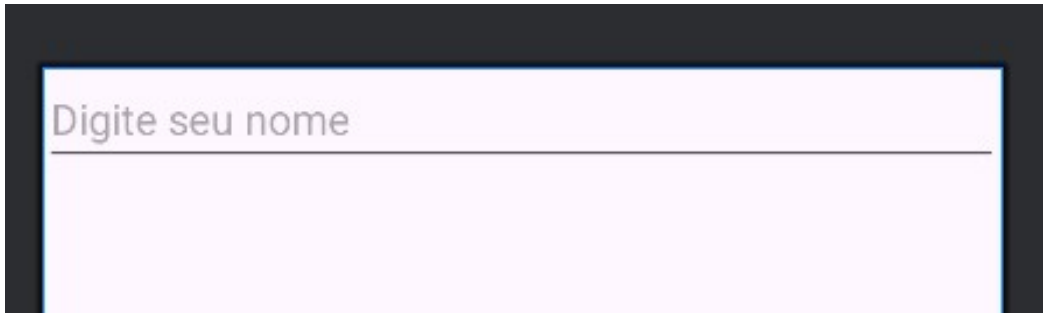
Em resumo, podemos apontar que o EditText se trata de um elemento de interface do usuário utilizado para entrada de texto, possibilitando a digitação ou edição de texto em um aplicativo Android.

EditText

```
<EditText
android:id="@+id/editTextNome
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Digite seu nome"
android:inputType="text"
android:maxLines="1" />
```

```
EditText editText = findViewById(R.id.editTextNome);
editText.setHint("Digite o seu Nome!");
```

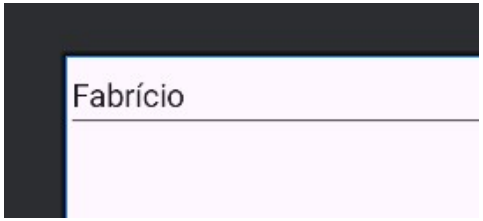
EditText



EditText

- **android:hint:** essa propriedade exibe um texto de dica (“hint”) no campo antes da entrada do usuário, orientando sobre o tipo de informação a ser inserida;
- **android:inputType:** define o tipo de entrada aceitável, possibilitando a configuração do campo para receber texto normal, senhas, e-mails, números, entre outros formatos;
- **android:maxLines:** especifica o número máximo de linhas visíveis permitidas no campo de texto.

EditText




```
EditText editText = findViewById(R.id.editTextNome);
Toast.makeText(this, "Seu nome é:" + editText.getText(),
Toast.LENGTH_SHORT).show();
```

Button

Função dos Botões

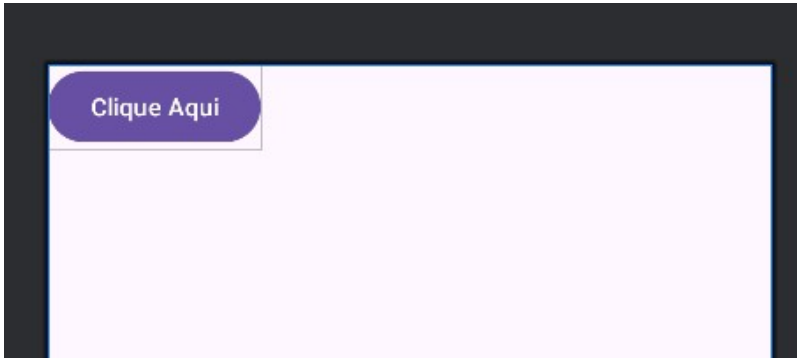
Os botões são fundamentais em interfaces digitais, permitindo que **usuários executem ações** como enviar formulários.

Personalização para Usabilidade

A personalização de botões pode melhorar a experiência do usuário, tornando a navegação mais intuitiva e acessível.

Button

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Clique Aqui" />
```



Button - Evento Dinâmico ou Estático

- **Estático**
  - Criamos um método **void** que recebe uma **view** por parâmetro e depois selecionamos ele no evento OnClick nas propriedades do botão

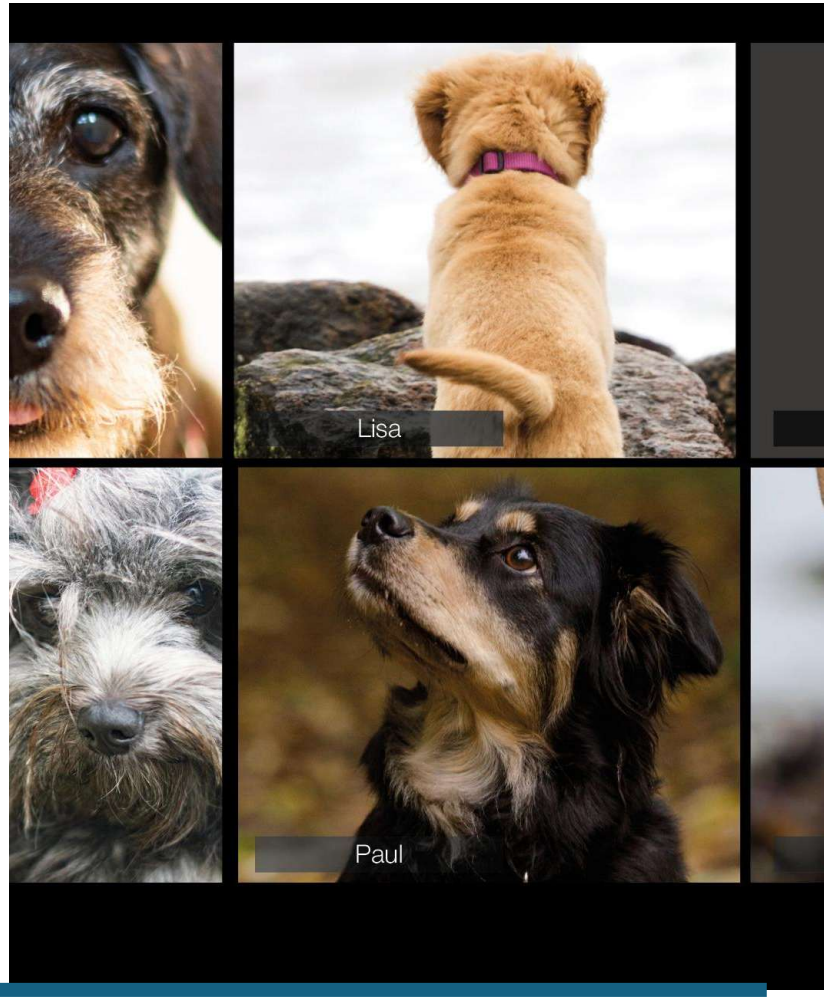
```
public void carregarActivityNova(View view)
{
   Intent intent = new Intent(MainActivity.this, NovaActivity.class);
   startActivity(intent); //esse método abre uma nova activity
}
```

Button - Evento Dinâmico ou Estático

- **Dinâmico**
  - Após criar um botão, adicionamos um Listner à esse botão através do método OnCreate.


```
<Button android:id="@+id/buttonEnviar"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Enviar" />
```

```
Button buttonEnviar = findViewById(R.id.buttonEnviar);
buttonEnviar.setOnClickListener(v -> {
    Toast.makeText(getApplicationContext(), "Botão pressionado!",
Toast.LENGTH_SHORT).show();
});
```



ImageView: Exibição de imagens

Uso do ImageView

O ImageView é uma ferramenta para exibir imagens dentro de aplicativos

Fontes de Imagem

O ImageView suporta várias fontes de imagem, permitindo carregar imagens de diferentes localizações, como URLs e arquivos locais.

Personalização da Apresentação

Com a personalização, é possível ajustar propriedades das imagens, como escala, rotação e bordas, melhorando a apresentação visual.

ImageView

```
<ImageView
android:id="@+id/imageViewLogo"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:src="@drawable/ic_launcher_foreground" />
```

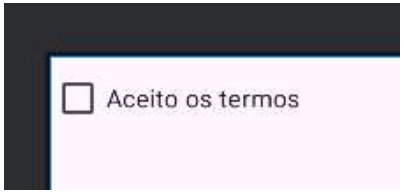
- ImageView imageViewLogo = findViewById(R.id.imageViewLogo);
- imageViewLogo.setImageResource(R.drawable.nova_imagem);

Checkbox: Caixas de seleção

Função das Caixas de Seleção

As caixas de seleção permitem que os usuários escolham uma ou mais opções de uma lista de maneira intuitiva e eficiente.

```
<CheckBox
   android:id="@+id/checkBox"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Aceito os termos" />
```




```
CheckBox checkBox = findViewById(R.id.checkBox);
checkBox.setOnCheckedChangeListener((buttonView,
isChecked) -> {
   if (isChecked) {
      Toast.makeText(getApplicationContext(), "Opção
selecionada!", Toast.LENGTH_SHORT).show();
   } else {
      Toast.makeText(getApplicationContext(), "Opção
desmarcada!", Toast.LENGTH_SHORT).show();
   }
});
```

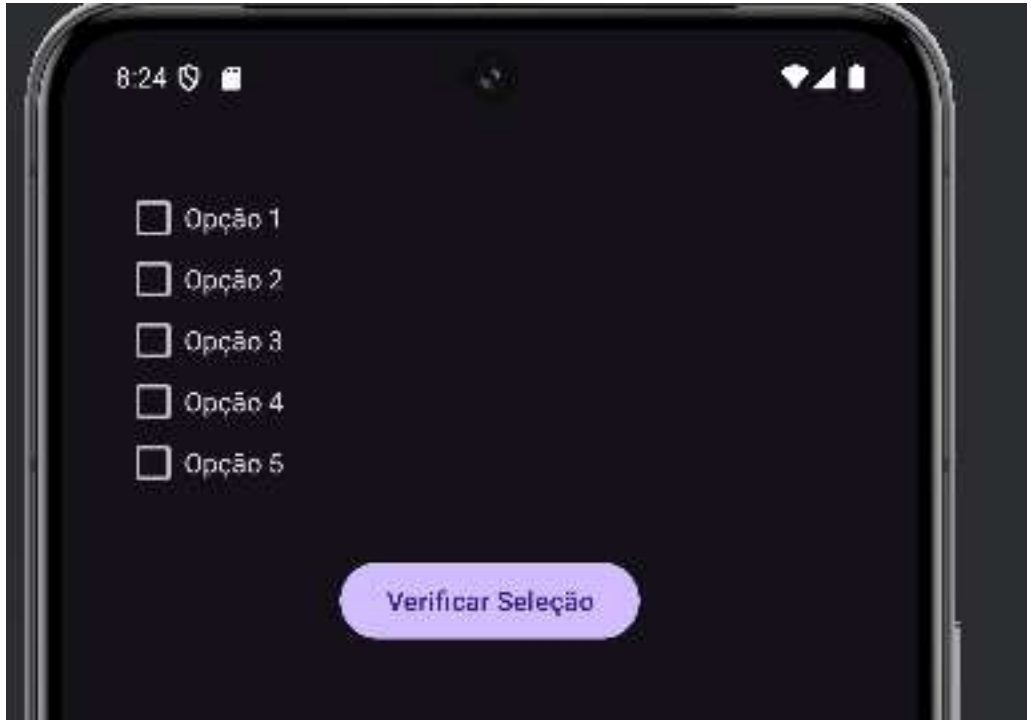
CheckBoxes Dinâmicos

```
private List<CheckBox> checkBoxList = new ArrayList<>();
```

```
<LinearLayout
    android:id="@+id/checkBoxContainer"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="16dp"
    android:layout_marginEnd="16dp"
    android:orientation="vertical"
    android:padding="16dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/btnCheck"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Verificar Seleção"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/checkBoxContainer" />
```

```
LinearLayout checkBoxContainer = findViewById(R.id.checkBoxContainer);
Button btnCheck = findViewById(R.id.btnCheck);

String[] opcoes = {"Opção 1", "Opção 2", "Opção 3", "Opção 4", "Opção 5"};

for (String opcao : opcoes)
{
    CheckBox checkBox = new CheckBox(this);
    checkBox.setText(opcao);
    checkBoxContainer.addView(checkBox);
    checkBoxList.add(checkBox);
}

btnCheck.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        StringBuilder selecionados = new StringBuilder("Selecionado: ");
        for (CheckBox checkBox : checkBoxList) {
            if (checkBox.isChecked()) {
                selecionados.append(checkBox.getText()).append(", ");
            }
        }

        if (selecionados.toString().equals("Selecionado: ")) {
            selecionados = new StringBuilder("Nenhuma opção selecionada!");
        } else {
            selecionados.setLength(selecionados.length() - 2);
        }

        Toast.makeText(getApplicationContext(), selecionados.toString(), Toast.LENGTH_SHORT).show();
    }
});
```

CheckBoxes Dinâmicos



```xml
<RadioGroup
  android:id="@+id/radioGroup"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:orientation="vertical">

  <RadioButton
    android:id="@+id/radio_opcao1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Opção 1"/>

  <RadioButton
    android:id="@+id/radio_opcao2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Opção 2"/>

  <RadioButton
    android:id="@+id/radio_opcao3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Opção 3"/>
</RadioGroup>
```

RadioButton: Botões de rádio

Seleção Única

Os botões de rádio permitem que os usuários escolham apenas uma opção entre várias, garantindo uma seleção única em formulários.

Uso em Formulários

Esses botões são frequentemente usados em formulários online e aplicativos para coletar informações do usuário.

Configuração e Gerenciamento

Vamos explorar como configurar adequadamente os botões de rádio e gerenciar as seleções feitas pelos usuários.

ScrollView

- O componente ScrollView é essencial quando se trata de exibir conteúdos mais extensos do que o espaço dispo nível na tela, permitindo a rolagem vertical ou horizontal de layouts que contenham dados além da área visível (Lecheta, 2010).

```
1  <ScrollView
2    android:layout_width="match_parent"
3    android:layout_height="match_parent">
4
5    <!-- Conteúdo rolável aqui -->
6
7  </ScrollView>
```

Diferenças - ScrollView e Scroll

CARACTERÍSTICA	SCROLLVIEW	SCROLL
Funcionalidade	Permite rolar um único item de layout	Ação de rolar o conteúdo na tela
Uso	Envolve o encapsulamento de um layout	A rolagem pode ser controlada por diferentes componentes
Controle de rolagem	Oferece controle total sobre a rolagem	Pode ser controlada por diferentes elementos de interface
Extensibilidade	Oferece flexibilidade limitada	Pode ser implementado em vários elementos de interface
Requisitos de memória	Pode consumir mais memória para conteúdos grandes	Depende do conteúdo e do tipo de elemento de interface
Implementação	É um componente específico e requer ser colocado em um layout	Pode ser implementado de diferentes maneiras conforme a necessidade

Spinner

```
<Spinner
   android:id="@+id/spinner"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"/>

<Button
   android:id="@+id/btnCheck"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Verificar Seleção"
   android:layout_marginTop="16dp"/>
```

res/values/strings.xml

```
<resources>
   <string name="app_name">Meu App</string>
   <string-array name="opcoes_spinner">
      <item>Opção 1</item>
      <item>Opção 2</item>
      <item>Opção 3</item>
      <item>Opção 4</item>
   </string-array>
</resources>
```

```
Spinner spinner = findViewById(R.id.spinner);
Button btnCheck = findViewById(R.id.btnCheck);

btnCheck.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View v) {
      String opcaoSelecionada = spinner.getSelectedItem().toString();
      Toast.makeText(ConsumirAPIActivity.this, "Selecionado: " +
opcaoSelecionada, Toast.LENGTH_SHORT).show();
   }
});
```

Exercício 1

- Crie um App onde o usuário informa o nome e a idade
- Mostre em tela a informação se ele é maior de idade ou não

Exercício 2

- Crie um App calculadora simples.
- Que recebe 2 valores e possua 1 botão para cada operação básica.

Exercício 3

- Faça um app que simule o cadastro de usuários para uma loja de roupas.
- O cadastro deve possuir:
  - Nome
  - Idade
  - UF
  - Cidade
  - Telefone
  - Email
  - RadioButton para o usuário selecionar o tamanho padrão de roupas
  - Uma listagem (checkboxes) para selecionar as cores de preferência dentro de uma lista de opções

Exercício 4

- Faça o usuário digitar o nome e mostre na tela de forma dinâmica
  1 checkbox para cada letra do nome

Exercício 5

- Em uma tela, adicione três CheckBox com as opções:
  - Receber notificações
  - Modo escuro
  - Lembrar login
- Adicione um Button chamado "Salvar Preferências".
- Ao pressionar o botão, exiba um Toast listando as opções marcadas.
- Se nenhuma opção for selecionada, exiba "Nenhuma preferência foi escolhida".

Exercício 6

- Construa uma Activity inicial com botões que permitem navegar entre as soluções de cada um dos exercícios.

Exercício 7

- Crie um repositório no GitHub e suba a aplicação desenvolvida que contenha a solução de todos os exercícios.