

Android e Java

Dr. Lucas Garofolo





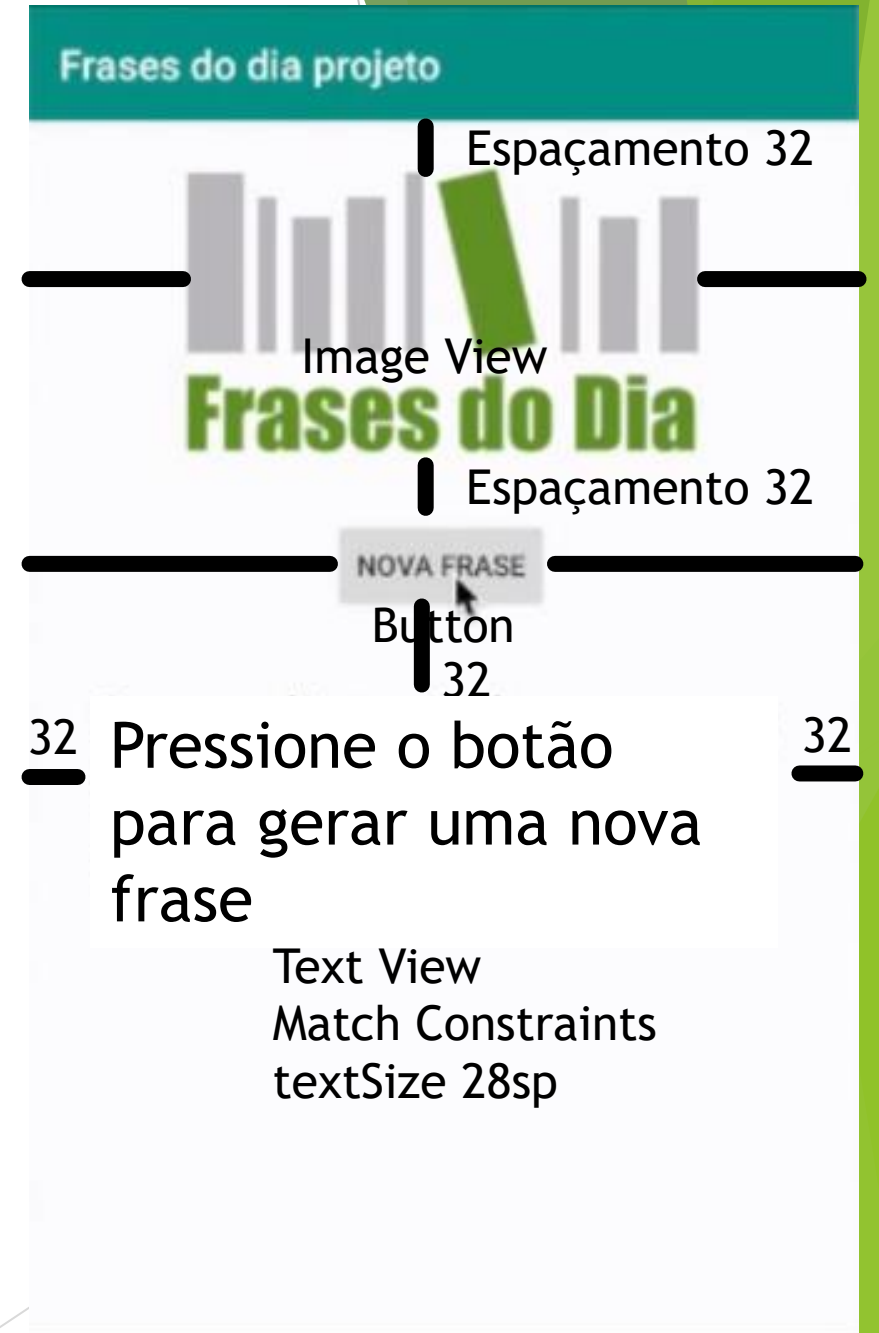
Frases do Dia

NOVA FRASE

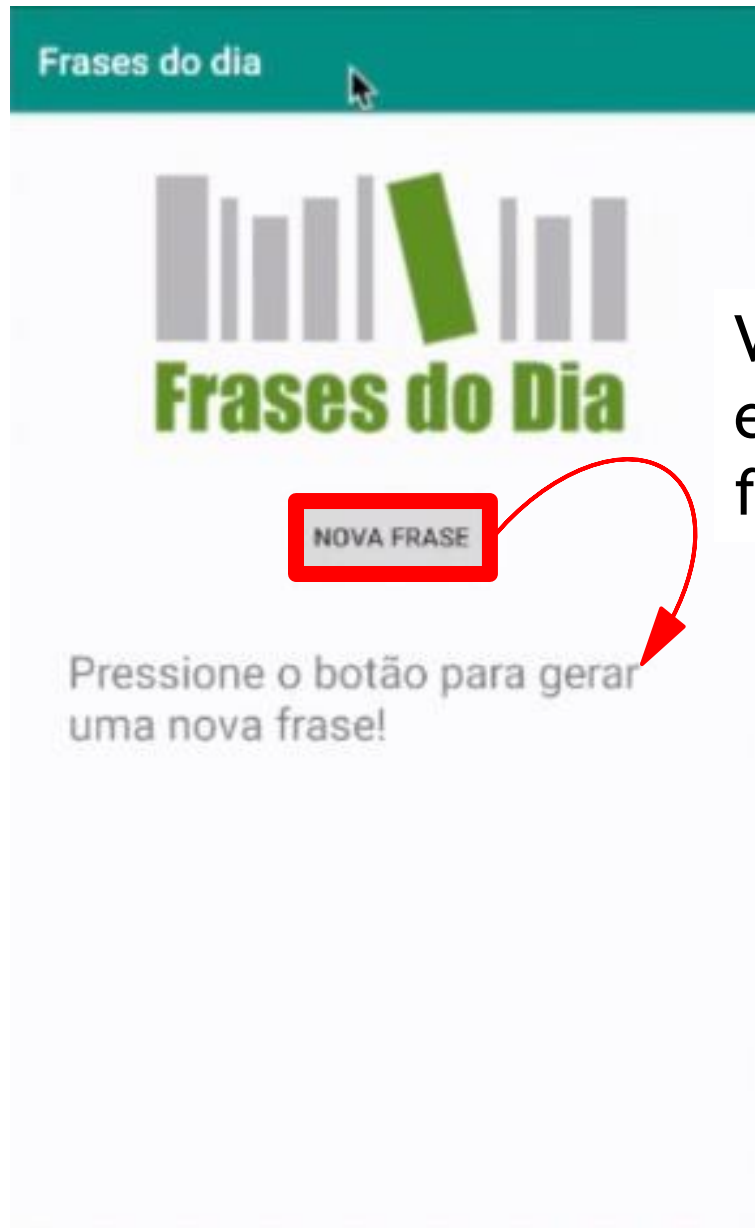
Que o dia seja leve, que a
tristeza seja breve e que
o dia seja feliz.

Vamos começar criando nosso app!

- ▶ Nosso app será o Frases do Dia
- ▶ Instruções de layout ao lado:
- ▶ Adicione no Manifest
`android:screenOrientation="sensor"`
- ▶ Adicione uma ScrollView no seu projeto novo



Agora vamos programar!



Vamos clicar no botão e o botão alterará a frase do TextView

Vamos programar o método

- No MainActivity.java, crie o método gerarNovaFrase:

```
public void gerarNovaFrase(View view) {  
    TextView texto = findViewById(R.id.textView);  
    texto.setText("Lucas");  
}
```

- No activity_main.xml adicione a ação do botão:

```
<Button  
    android:id="@+id/button"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_marginTop="280dp"  
    android:text="Button"  
    android:onClick="gerarNovaFrase"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.498"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/textView" />
```

- ▶ Já adicionamos a ação no nosso botão, mas nosso programa está fazendo o que nós gostaríamos que fizesse?
- ▶ Nosso programa precisa ter um banco de dados de frases.
- ▶ O Java possui alguma estrutura que nós conseguiríamos guardar várias frases que a cada clique mostrasse uma frase diferentes?

```
public void gerarNovaFrase(View view) {  
  
    String[] frases = {  
        "Frase 1",  
        "Frase 2",  
        "Frase 3",  
        "Frase 4"  
    };  
  
    TextView texto = findViewById(R.id.textView);  
    texto.setText(frases[0]);  
}
```

Array

- ▶ Faça o teste e veja o que acontece.
- ▶ A frase mudou?
- ▶ Não. Nós configuramos a frase da posição 0.
- ▶ Precisamos de alguma variável que modifique o índice do array frases.



DESAFIO

Final

- Uma vez, escolhido o método que utilizará para alterar as frases, coloque no mínimo 10 frases motivacionais no nosso app.
- Sugeri 4 delas no classroom da disciplina.



Atividade 1

- ▶ Seguindo a metodologia que utilizamos nas últimas duas aulas, desenvolva um aplicativo com um dos seguintes temas:
 - ▶ **Pensamento do Dia:** onde os usuários recebem diariamente um pensamento ou reflexão sobre diversos temas, como amor, felicidade, sucesso, etc.
 - ▶ **Fatos Interessantes:** um aplicativo que fornece um fato interessante ou curioso a cada dia, para estimular o aprendizado e a curiosidade dos usuários.
 - ▶ **Dicas de Saúde e Bem-Estar:** oferece dicas diárias relacionadas à saúde, nutrição, exercícios físicos, bem-estar mental, entre outros aspectos relacionados à qualidade de vida.
 - ▶ **Receitas do Dia:** um aplicativo que apresenta uma nova receita culinária todos os dias, incentivando os usuários a experimentarem novos pratos e ingredientes.

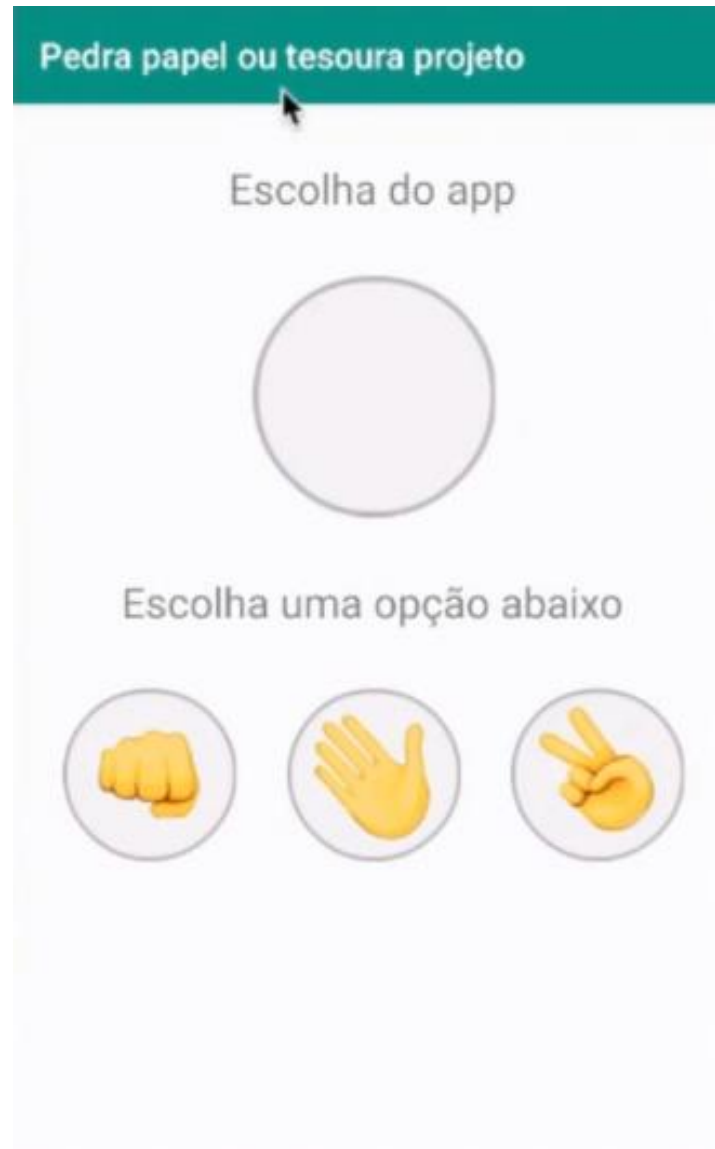
Jogo simples

Pedra, papel e tesoura

JOKENPO

O nosso app irá funcionar da seguinte forma:

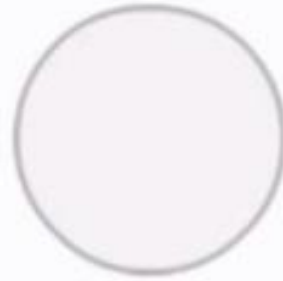
- Nós iremos clicar numa das opções de pedra, papel e tesoura.
- O app irá escolher uma opção de pedra papel e tesoura de forma aleatória.
- E o aplicativo irá nos informar se ganhamos, perdemos ou empatados.
- Daí o jogo recomeça.



Layout

Pedra papel ou tesoura projeto

Escolha do app



Escolha uma opção abaixo



Layout

JOKENPO

- Elementos centralizados.
- Espaçamento superior sempre de 32
- Utilize **CHAINS** para alinhar os 3 elementos inferiores
- Elemento principal com 100dp de height e width



Agora vamos pro Java

- ▶ No MainActivity.java precisamos criar um método para poder comparar a nossa jogada com a jogada do computador.
- ▶ E três métodos para reconhecer a nossa jogada:
 - ▶ Quando pedra for jogado
 - ▶ Quando papel for jogado
 - ▶ Quando tesoura for jogado

Agora vamos pro Java

► No MainActivity.java:

```
public void comparaSelecao(String selecaoUsuario) {  
  
}
```

```
public void selecaoPedra(View view) {  
    this.comparaSelecao( selecaoUsuario: "pedra" );  
}
```

► Implemente os outros dois métodos para a seleção de Tesoura e Papel!

Implementando o método comparaSelecao

- ▶ Precisamos implementar um número aleatório de 0 a 2.
- ▶ Criar um array com os seguintes textos: pedra, papel ou tesoura, que compreende as posições 0, 1 e 2 do sorteio de número aleatório.
- ▶ Criar um Switch Case para colocar a figura correta no resultado da tela de acordo com o sorteio do número aleatório.

Implementando o método comparaSelecao

- Precisamos implementar um número aleatório de 0 a 2.

```
public void comparaSelecao(String selecaoUsuario) {  
    Integer numero = new Random().nextInt(bound: 3);  
}
```

- Criar um array com os seguintes textos: pedra, papel ou tesoura, que compreende as posições 0, 1 e 2 do sorteio de número aleatório.

```
public void comparaSelecao(String selecaoUsuario) {  
    Integer numero = new Random().nextInt(bound: 3);  
    String[] opcoes = {"pedra", "papel", "tesoura"};  
    String opcaoApp = opcoes[numero];  
}
```


Implementando o método comparaSelecao

- Criar um Switch Case para colocar a figura correta no resultado da tela de acordo com o sorteio do número aleatório.

```
public void comparaSelecao(String selecaoUsuario) {  
    Integer numero = new Random().nextInt(bound: 3);  
    String[] opcoes = {"pedra", "papel", "tesoura"};  
    String opcaoApp = opcoes[numero];  
  
    ImageView jogadaApp = findViewById(R.id.imageView);  
    switch (opcaoApp) {  
        case "pedra":  
            jogadaApp.setImageResource(R.drawable.pedra);  
            break;  
    }  
}
```

Implemente o código
para papel e tesoura.

Implementando o método comparaSelecao

- ▶ Agora precisamos implementar um IF para verificar quem ganhou a rodada.

TABELA VERDADE

Usuário/App	Tesoura	Pedra	Papel
Tesoura			
Pedra			
Papel			

Implementando o método comparaSelecao

- Implemente o resultado da tabela verdade dentro do If Else para incluir a lógica de ganhador e perdedor.

```
TextView textoResultado = findViewById(R.id.textView);  
if (//implemente quando o usuário ganhar) {  
    textoResultado.setText("Você ganhou!");  
}  
else if (//implemente quando o usuário perder) {  
    textoResultado.setText("Você perdeu!");  
}  
else {  
    textoResultado.setText("Empatou");  
}
```

Desafio final

1. **Coin Flip (Cara ou Coroa):** Um aplicativo onde os usuários podem simular uma jogada de cara ou coroa, onde a moeda é lançada aleatoriamente e o resultado é exibido.
2. **Dado Virtual:** Um aplicativo que simula um dado virtual, permitindo que os usuários rolem o dado e recebam um número aleatório como resultado.
3. **Roleta da Sorte:** Um aplicativo onde os usuários podem girar uma roleta virtual e ver em qual opção ela para, como uma forma de sorteio aleatório.
4. **Baralho Virtual:** Um aplicativo que simula um baralho de cartas, permitindo que os usuários embaralhem as cartas e recebam uma carta aleatória.