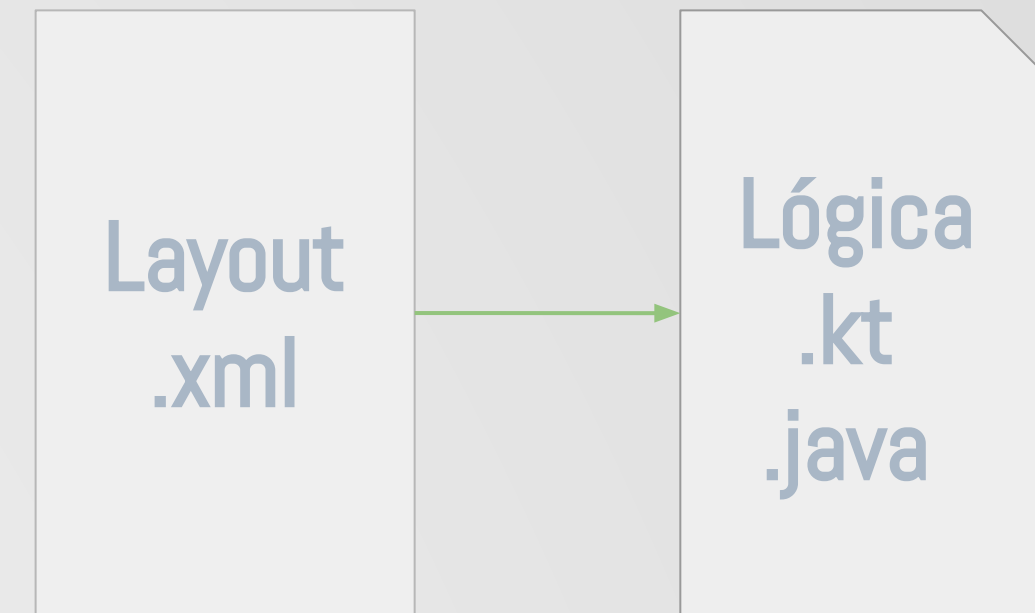

Definições de Layout



Android Layout

- Define a estrutura da interface gráfica
 - Como os elementos se comportam e se relacionam
- Layout e codificação são tratados como itens separados
- Todos os elementos dentro do layout respeitam a hierarquia de View e ViewGroup



View

Geralmente representam elementos na interface que o usuário possa interagir.

Button, EditText, Switch, ImageView, CheckBox

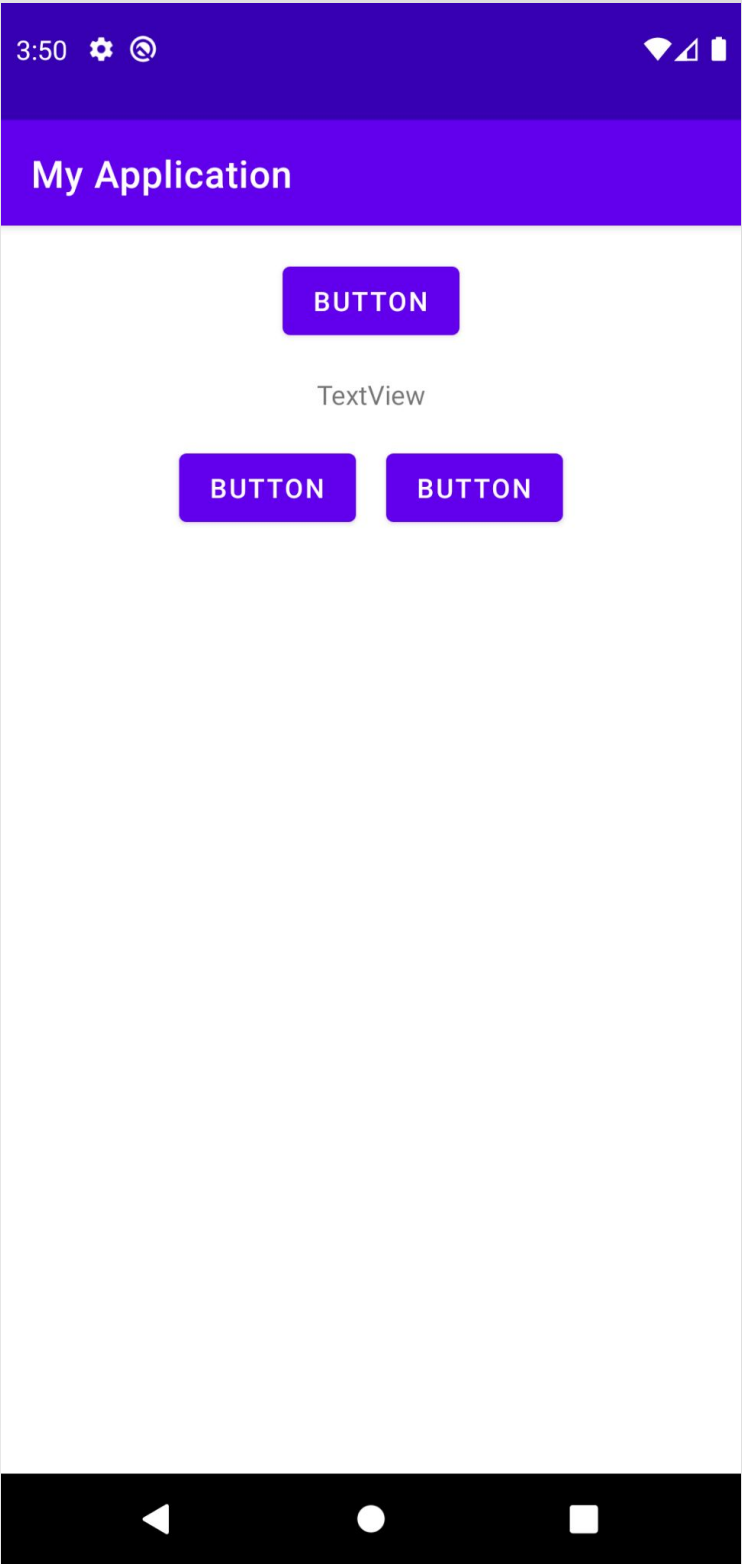
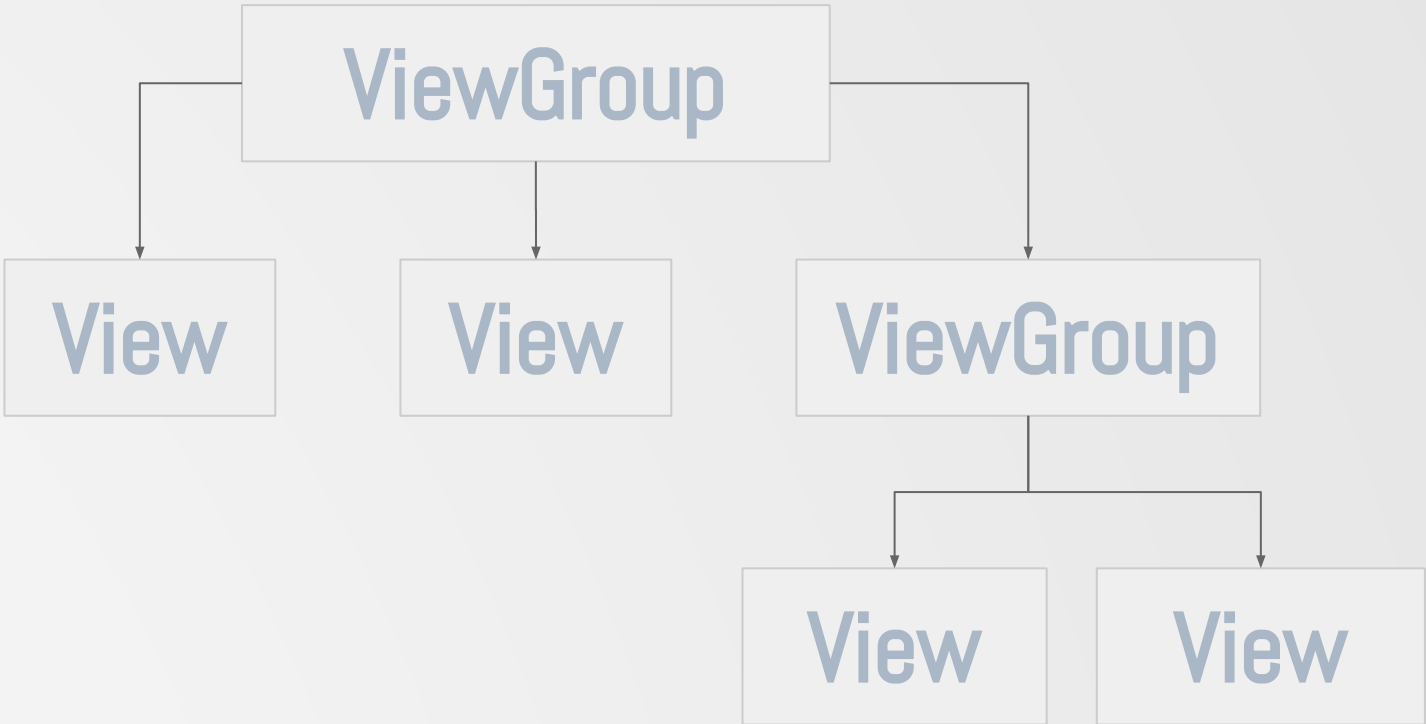
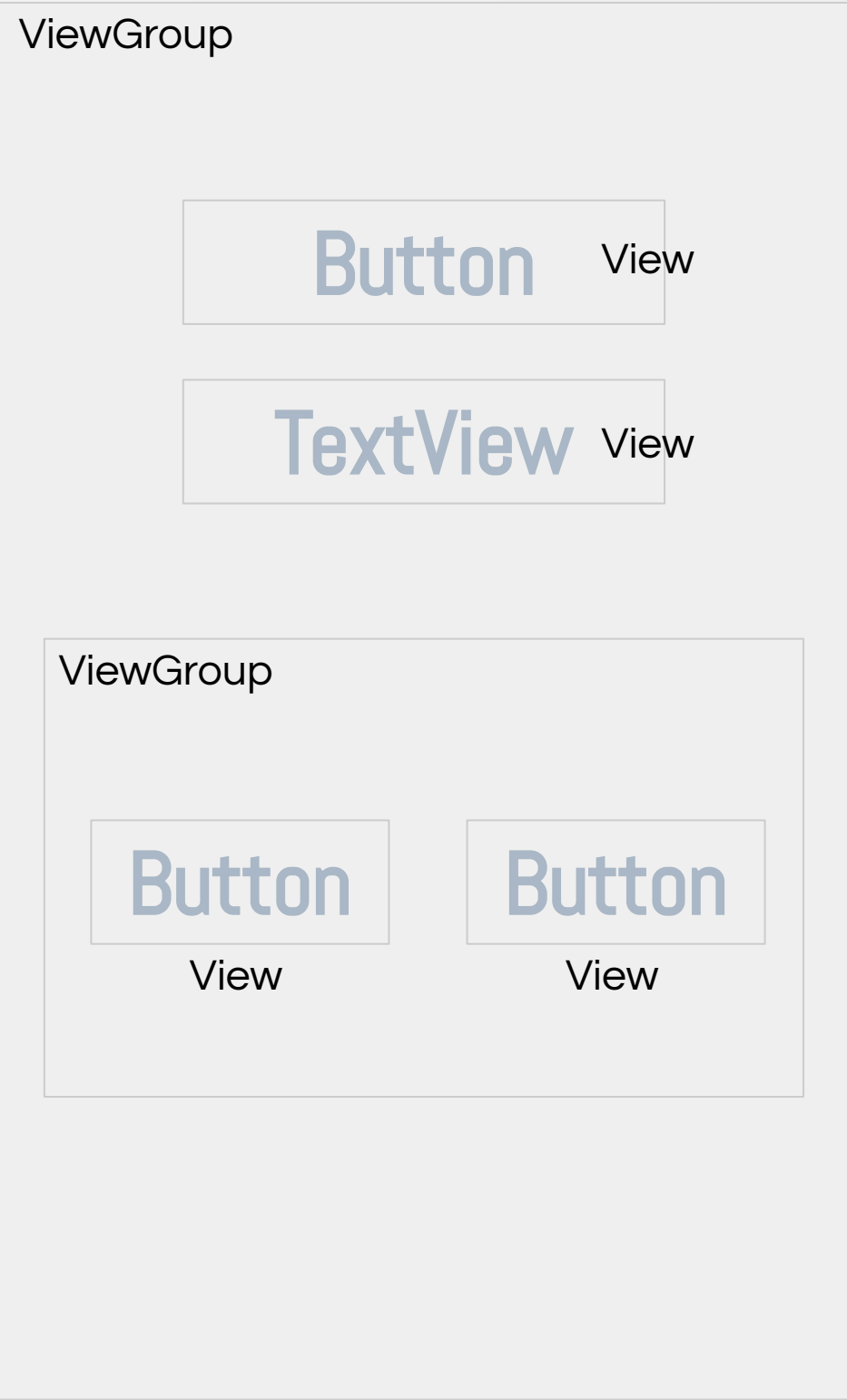
ViewGroup

Elementos de layout que definem o comportamento de outros elementos. Um agrupador de elementos de interface.

LinearLayout, RelativeLayout, ConstraintLayout, RecyclerView

- Um elemento de layout pode ser definido via código (dinamicamente) ou como XML (estaticamente)

Android Layout



Boas práticas - Estrutura de Layout

- A quantidade de elementos influencia no processamento e tempo de carregamento da tela
 - Quanto mais elementos e mais níveis, mais custo para processamento

```
<LinearLayout>

    <Button />
    <TextView />

    <LinearLayout>
        <Button />
        <Button />
    </LinearLayout>

</LinearLayout>
```

Correto

```
<LinearLayout>
    <LinearLayout>
        <Button />
    </LinearLayout>

    <LinearLayout>
        <TextView />
    </LinearLayout>

    <LinearLayout>

        <LinearLayout>
            <Button />
        </LinearLayout>
        <LinearLayout>
            <Button />
        </LinearLayout>

    </LinearLayout>
</LinearLayout>
```

Incorreto

Boas práticas - Tamanhos

- **Usar as medidas recomendadas**

- Tamanho de fonte devem usar a medida "sp" (Scale independent pixels - Escala independente de pixels)
- Tamanho de elementos, margins, paddings, etc devem usar "dp" (Density Independent pixels - Densidade independente de pixels)

```
android:textSize="12sp"  
android:layout_margin="12dp"
```

- **Largura e altura também podem ser definidas relativas ao elemento pai/mãe ou ao próprio elemento**

- match_parent - Quando um elemento possui este valor no atributo largura ou altura (width ou height), seu tamanho será igual o tamanho do elemento pai/mãe (elemento exterior, nível acima)
- wrap_content - Quando um elemento possui este valor no atributo largura ou altura (width ou height), seu tamanho será o suficiente para acomodar seu conteúdo

```
android:layout_width="match_parent"  
android:layout_height="wrap_content"
```

Boas práticas - XML

- Elementos são sempre escrito com “self closing tags”

```
<Button />
```

Correto

```
<Button></Button>
```

Incorreto

- Elementos são case sensitive

```
<Button />
```

Correto

```
<button />
```

Incorreto