

Programación en Android II

Layouts

Programación Multimedia y Dispositivos Móviles
2º Técnico en Desarrollo de Aplicaciones Multiplataforma

Antes de empezar



Antes de empezar



Antes de empezar



¿Qué vamos a ver?

1

Introducción

2

LinearLayout

3

TableLayout

4

FrameLayout

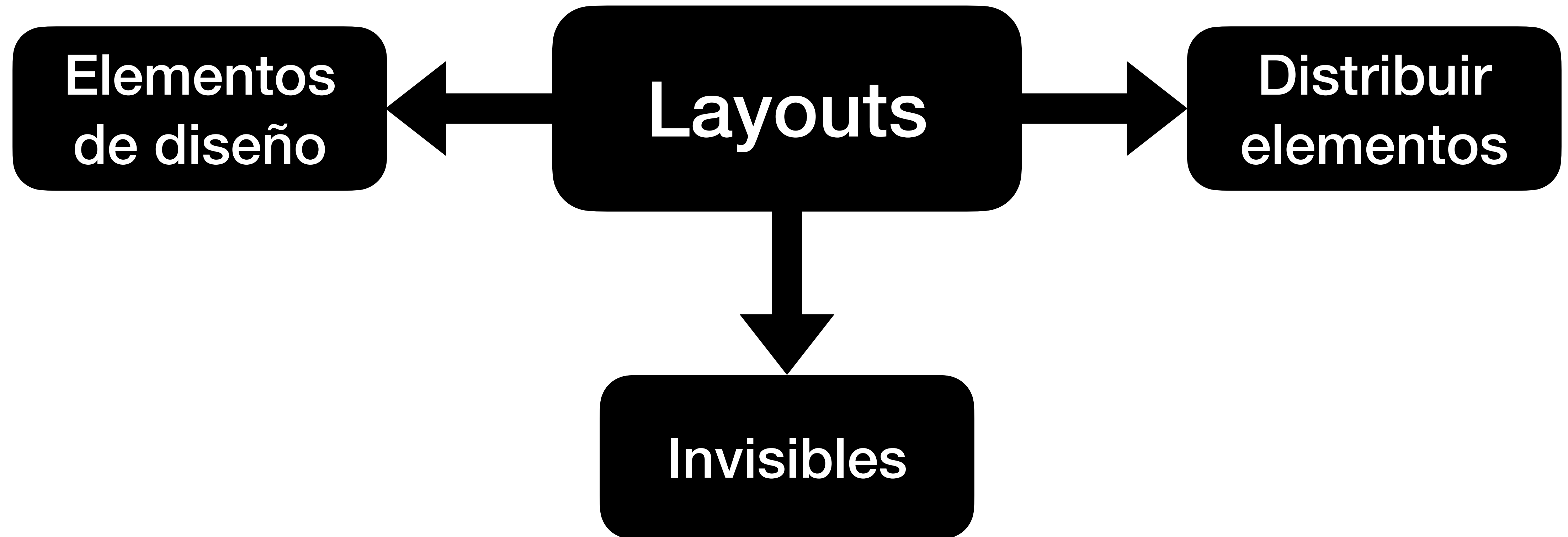
5

ConstraintLayout

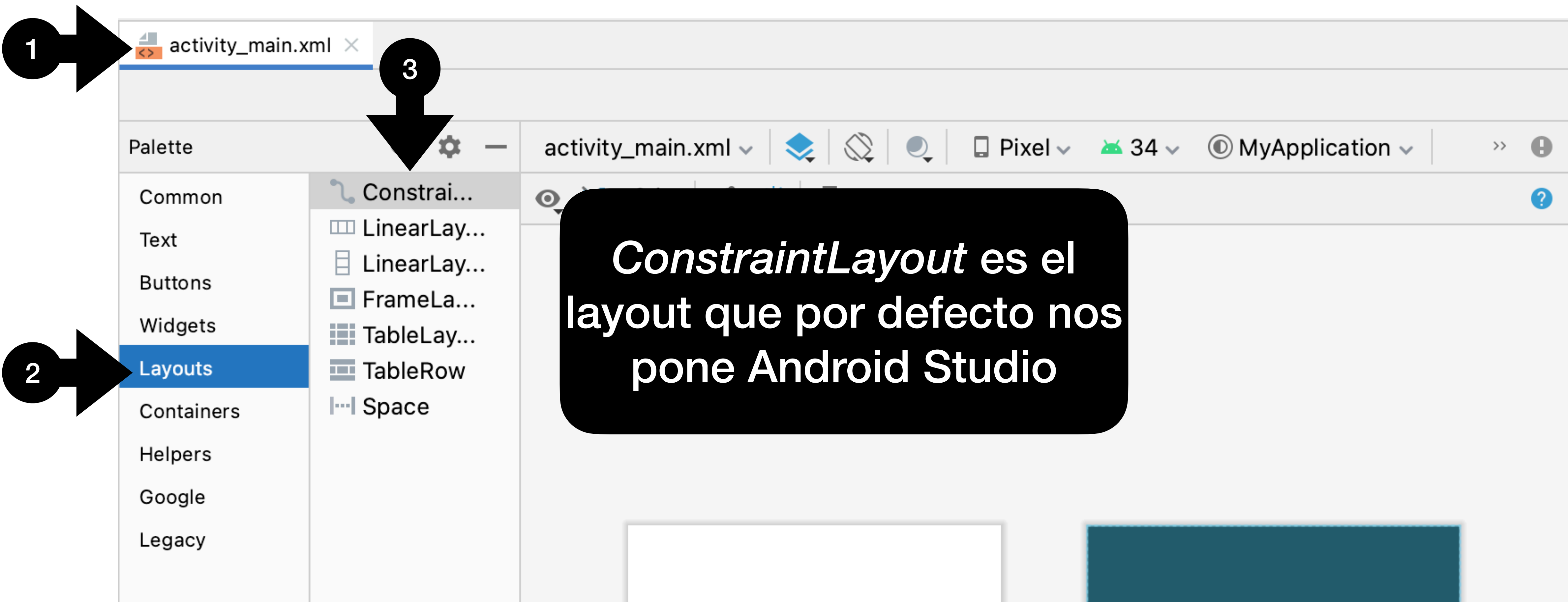
6

Pestañas

1 Introducción



1 Introducción



Utilizaremos:

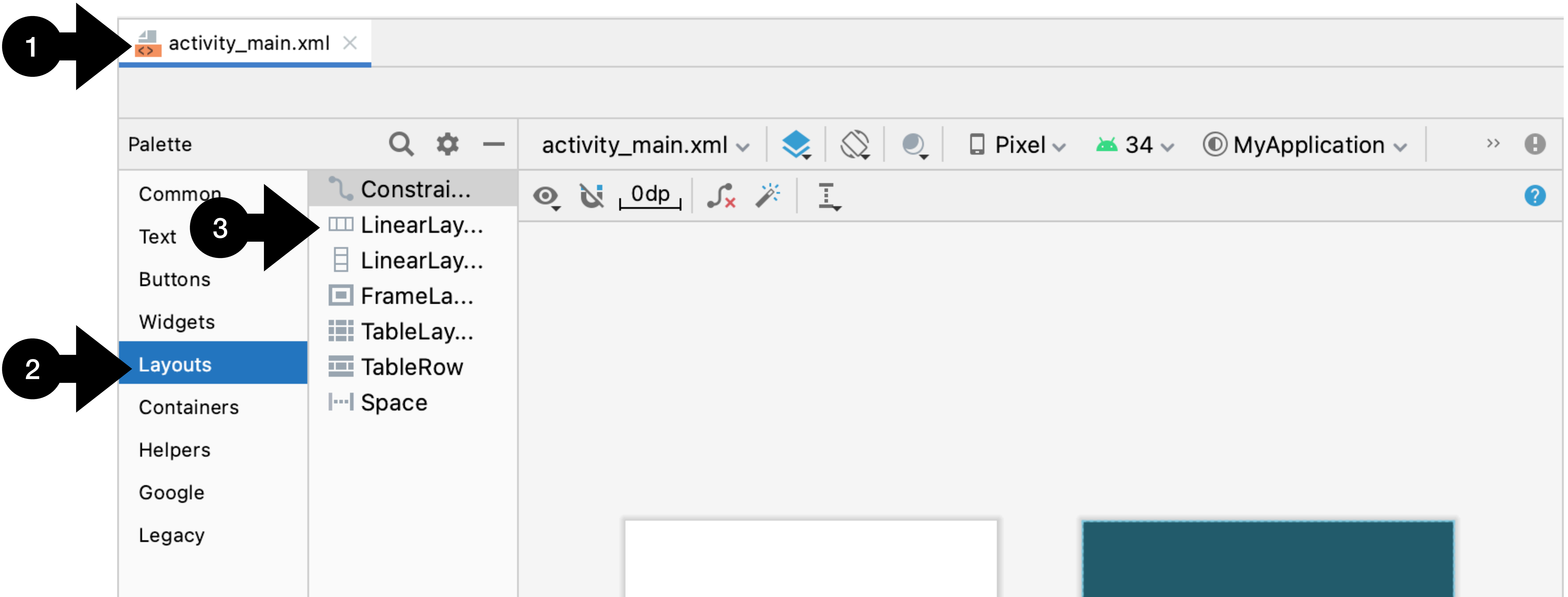
TableLayout

LinearLayout

FrameLayout

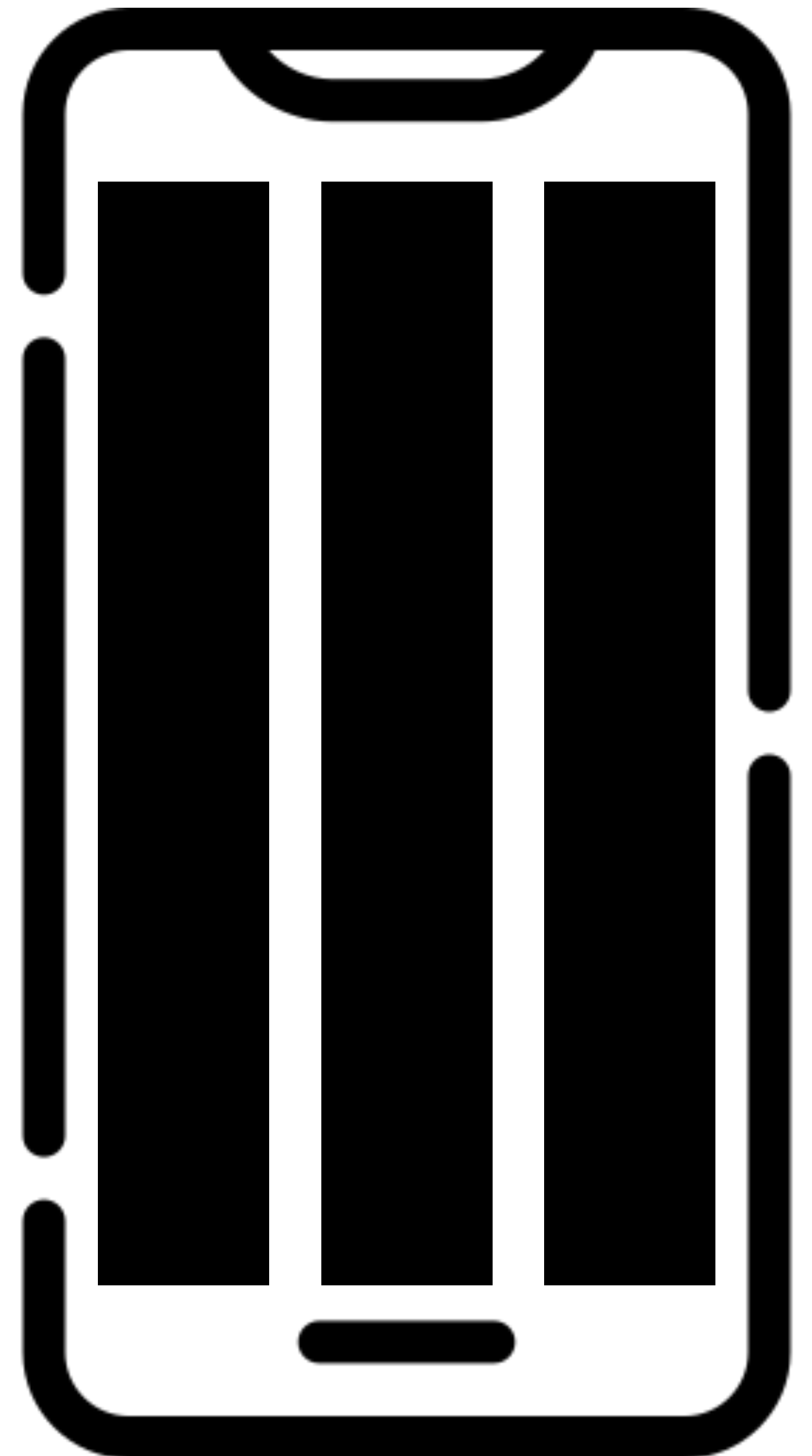
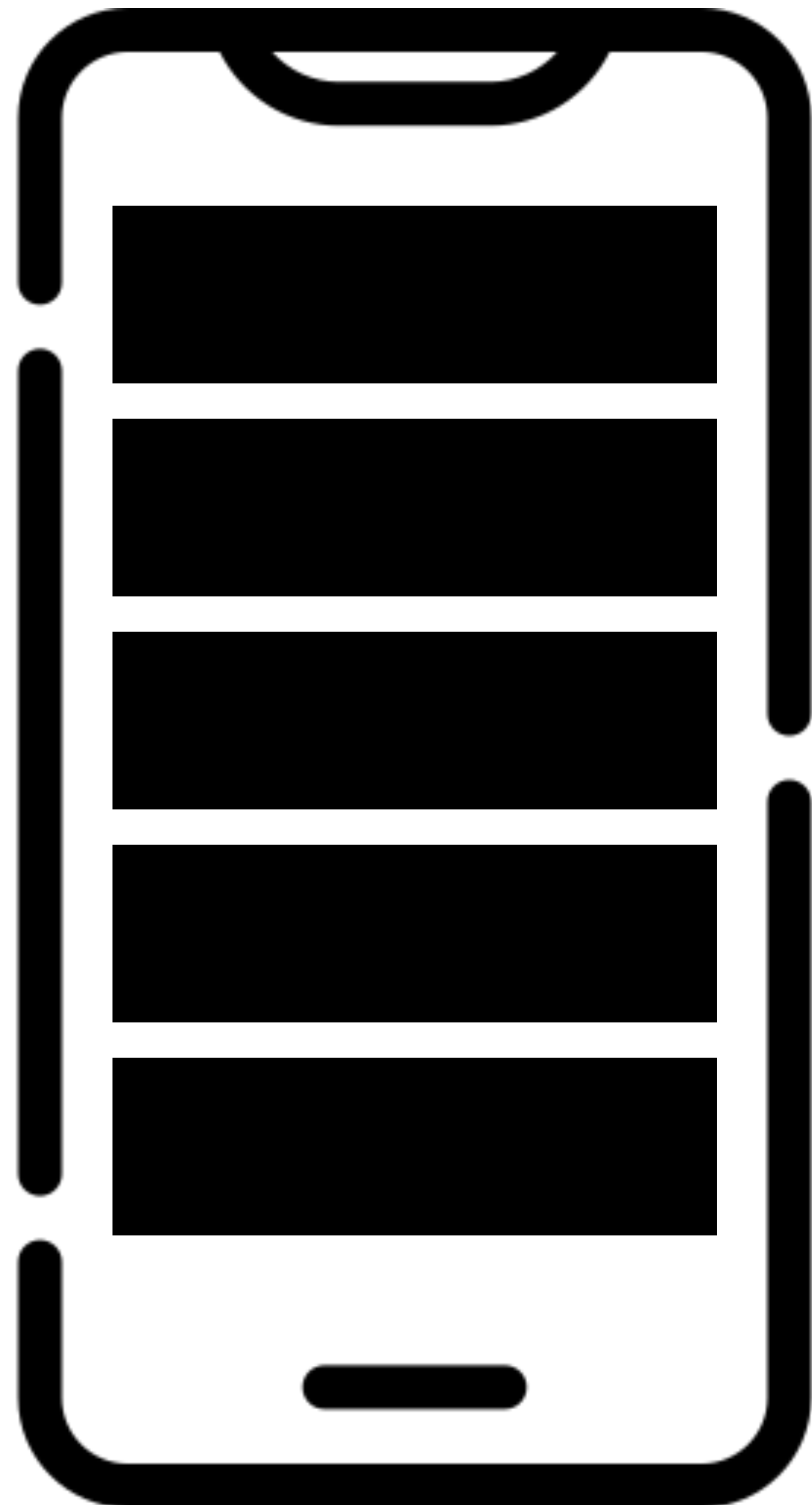
ConstraintLayout

2 LinearLayout

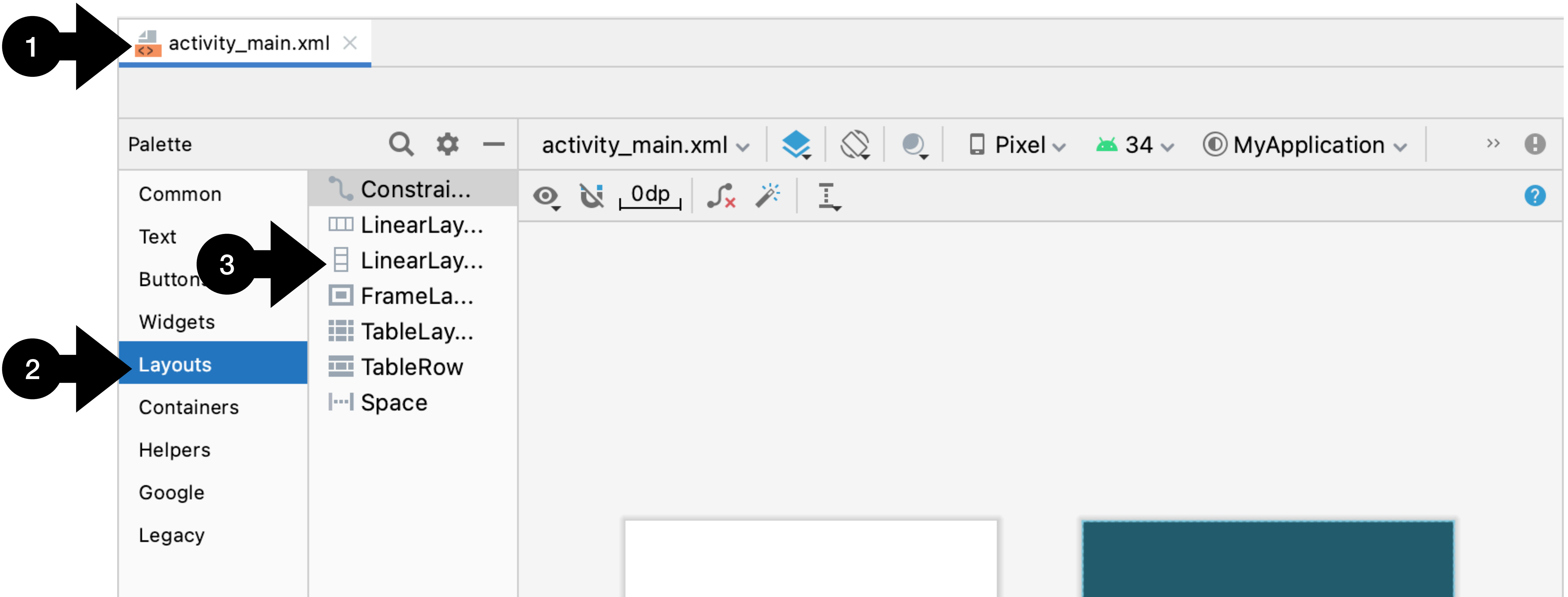


2 LinearLayout

Divide los
elementos de
forma
consecutiva



2 LinearLayout



2 LinearLayout

activity_main.xml

Code Split Design

activity_main.xml Pixel 34 MyApplication

1

2

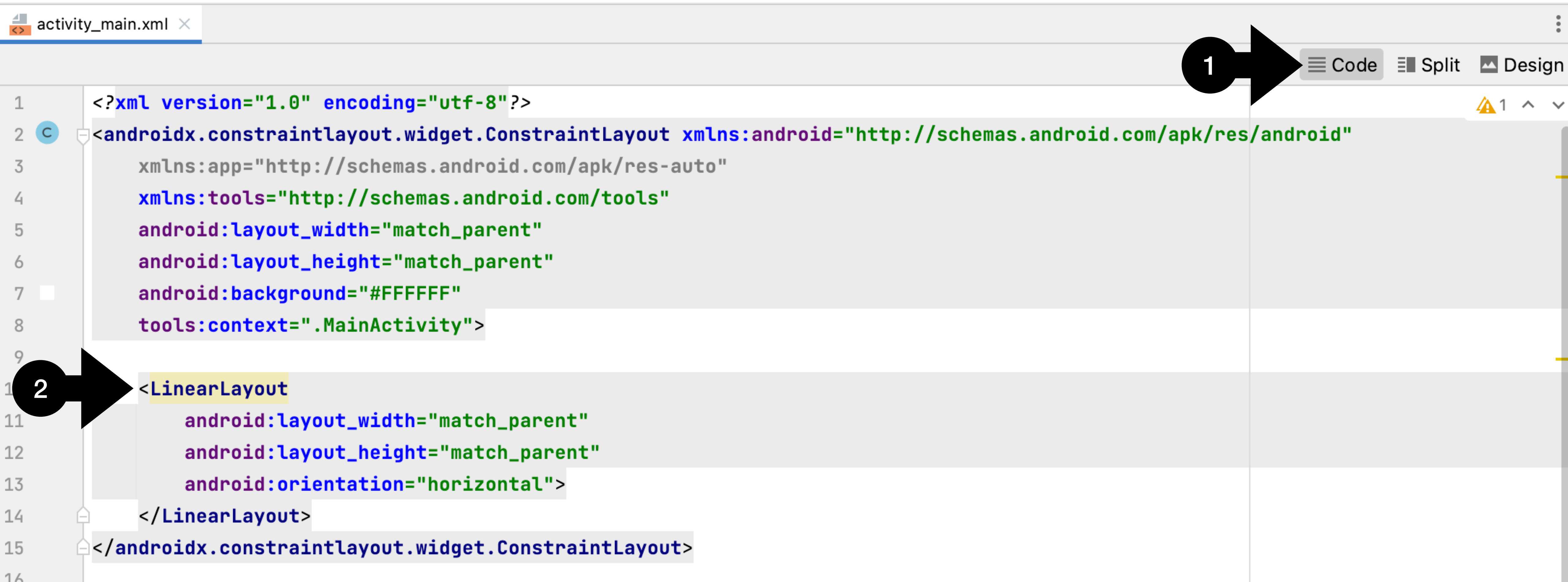
3

Attributes

ConstraintLayout <unnamed>

id	
Declared Attributes	
layout_width	match_parent
layout_height	match_parent
background	#FFFFFF
context	.MainActivity
Layout	
Transforms	
Common Attributes	
All Attributes	
accessibilityLiv...	

2 LinearLayout



2 LinearLayout

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="horizontal">  
</LinearLayout>
```

android:orientation="horizontal"

android:orientation="vertical"

2 LinearLayout

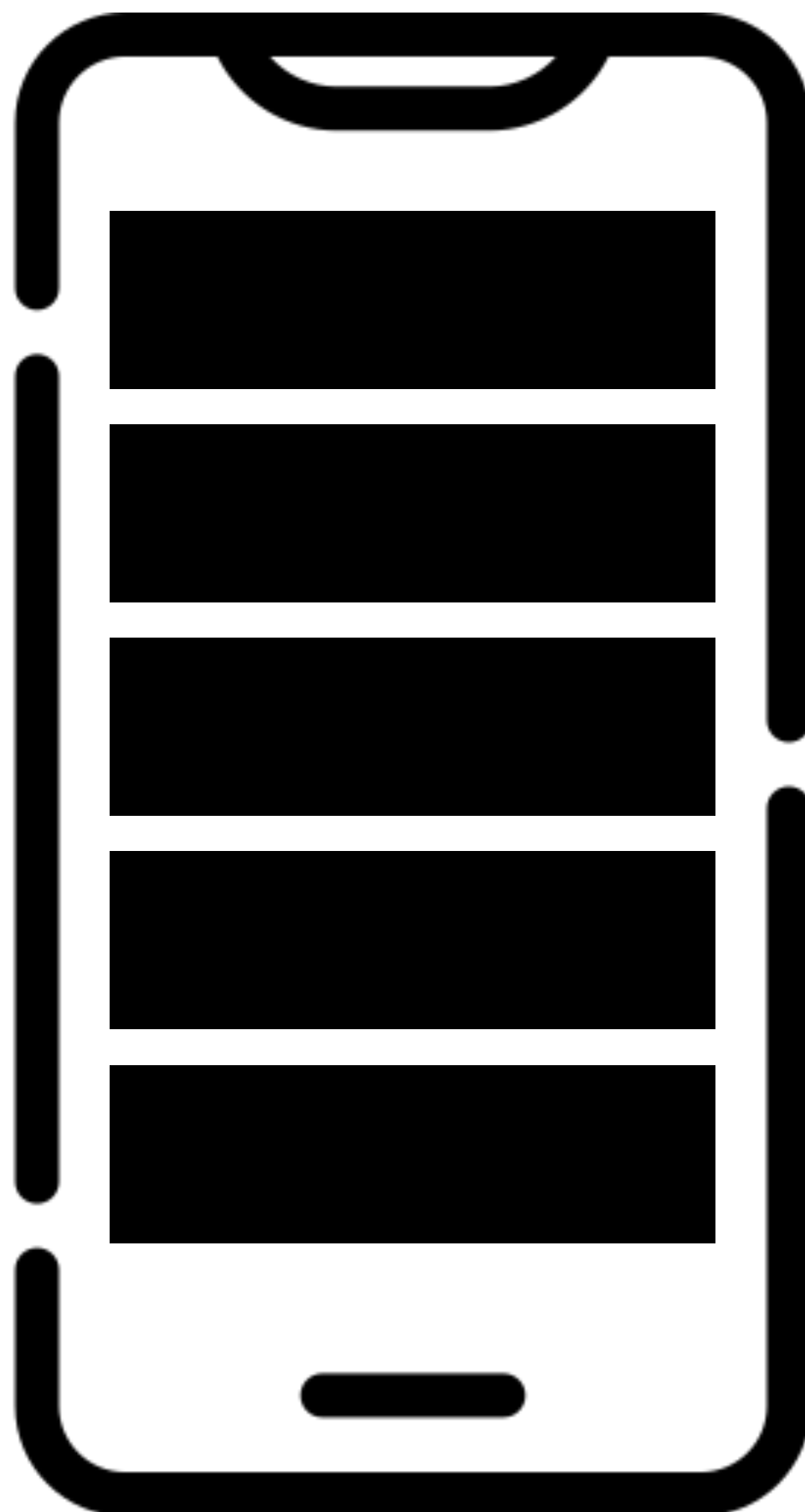
The screenshot shows the Android Studio IDE with the following components:

- Palette:** The 'Layouts' category is selected. The 'LinearLayout' widget is highlighted.
- Component Tree:** The 'LinearLayout (horizontal)' widget is selected.
- Design View:** The layout is shown with a white background and a dark blue rectangular widget. The widget is labeled with a '2' and an arrow pointing to the Attributes panel.
- Attributes Panel:** The 'LinearLayout' widget is selected. The 'Declared Attributes' section shows the following attributes:

Attribute	Value	Type
layout_width	match_parent	Dimension
layout_height	match_parent	Dimension
orientation	horizontal	Enum

2 LinearLayout

Vertical

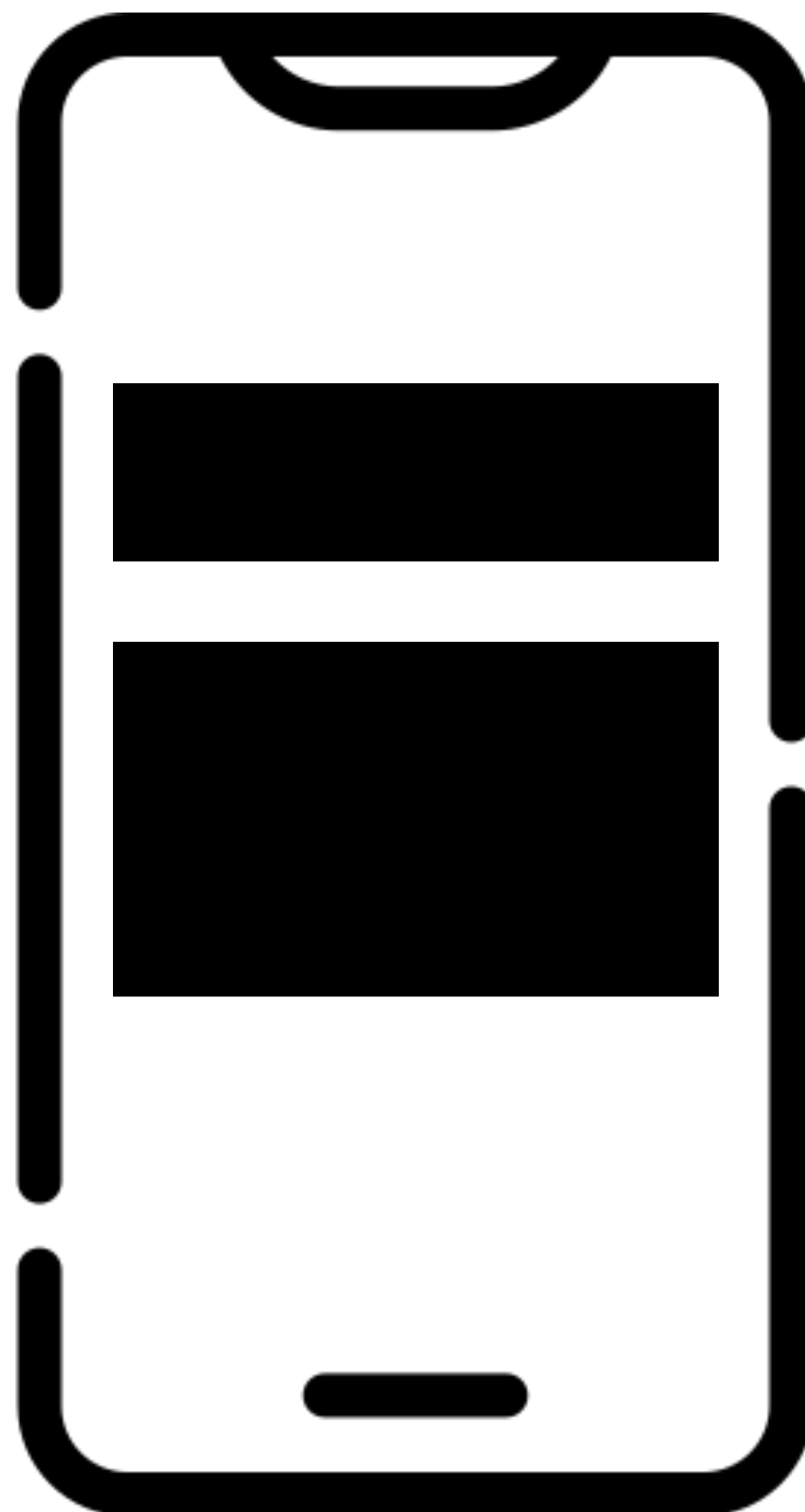


Horizontal



2 LinearLayout

Vertical



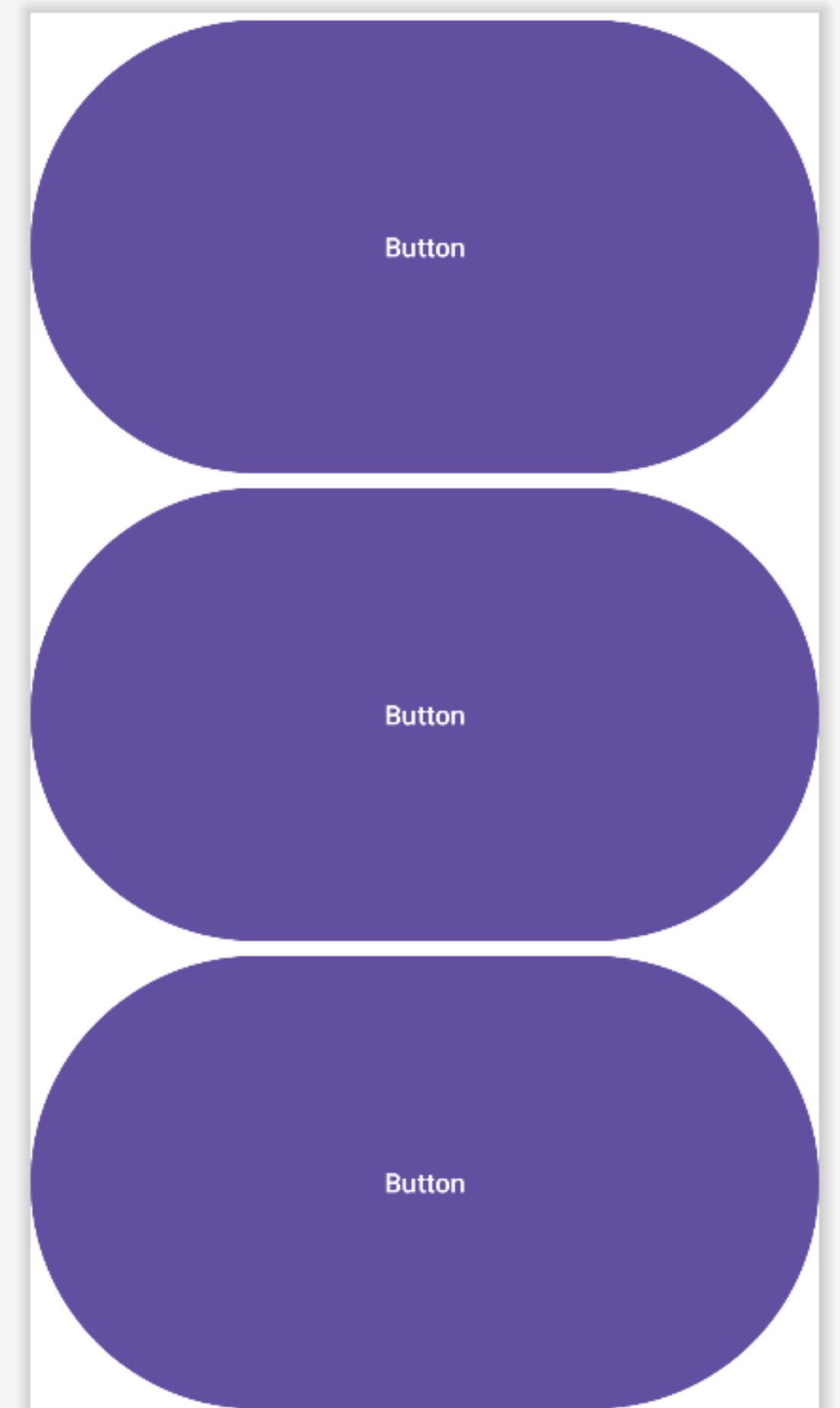
Podemos definir que
utilicen tamaños
concretos
proporcionales

`android:layout_weight`

Horizontal



```
14
15
16     <Button
17         android:id="@+id/button"
18         android:layout_width="match_parent"
19         android:layout_height="wrap_content"
20         android:layout_weight="1"
21         android:text="Button" />
22
23     <Button
24         android:id="@+id/button2"
25         android:layout_width="match_parent"
26         android:layout_height="wrap_content"
27         android:layout_weight="1"
28         android:text="Button" />
29
30     <Button
31         android:id="@+id/button3"
32         android:layout_width="match_parent"
33         android:layout_height="wrap_content"
34         android:layout_weight="1"
35         android:text="Button" />
36 </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```



```
<Button
    android:id="@+id/button"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Button" />
```

```
<Button
    android:id="@+id/button2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="2"
    android:text="Button" />
```

```
<Button
    android:id="@+id/button3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Button" />
```

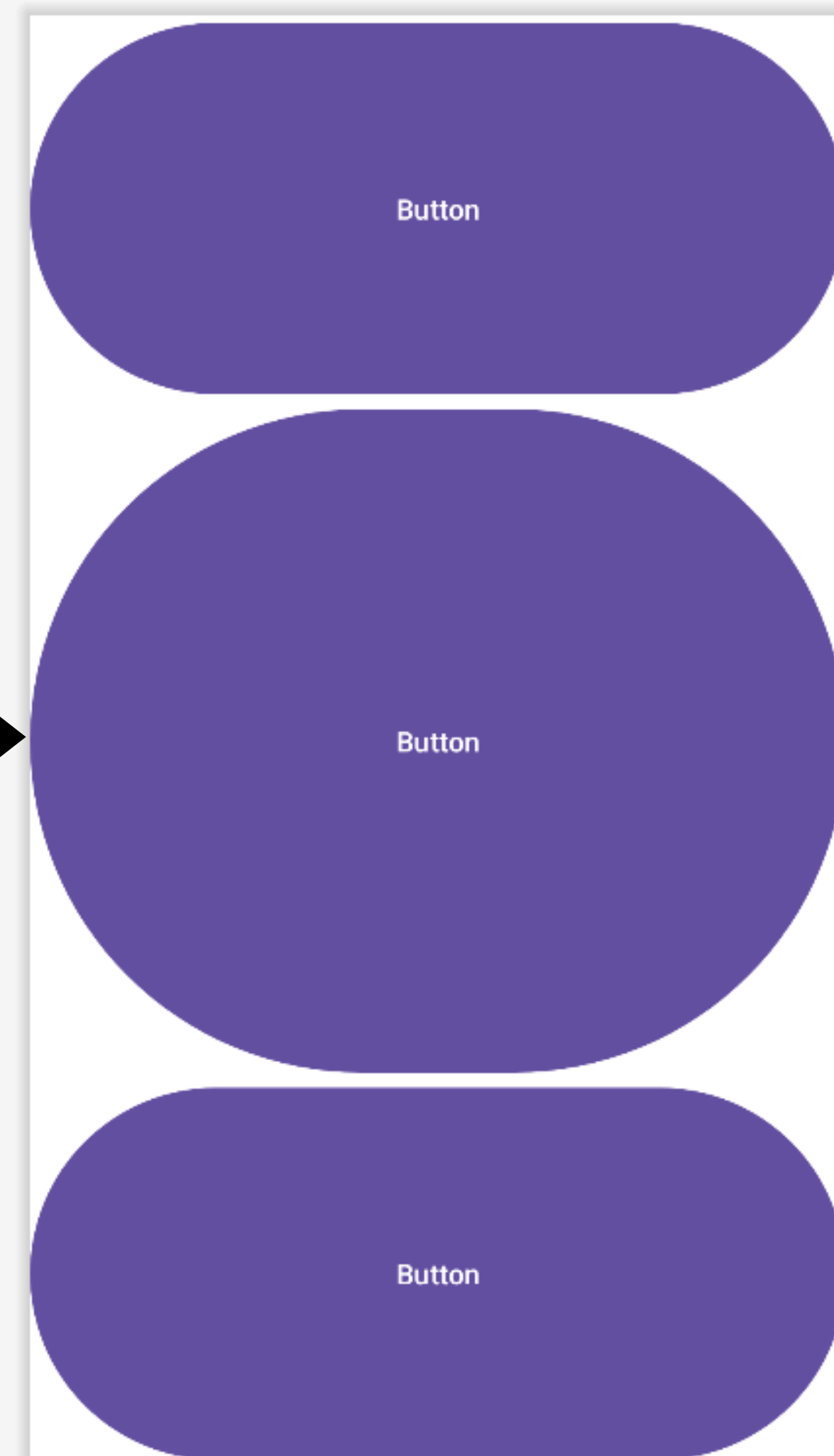
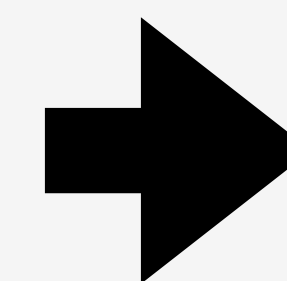
```
</LinearLayout>
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```


Palette


activity_main.xml

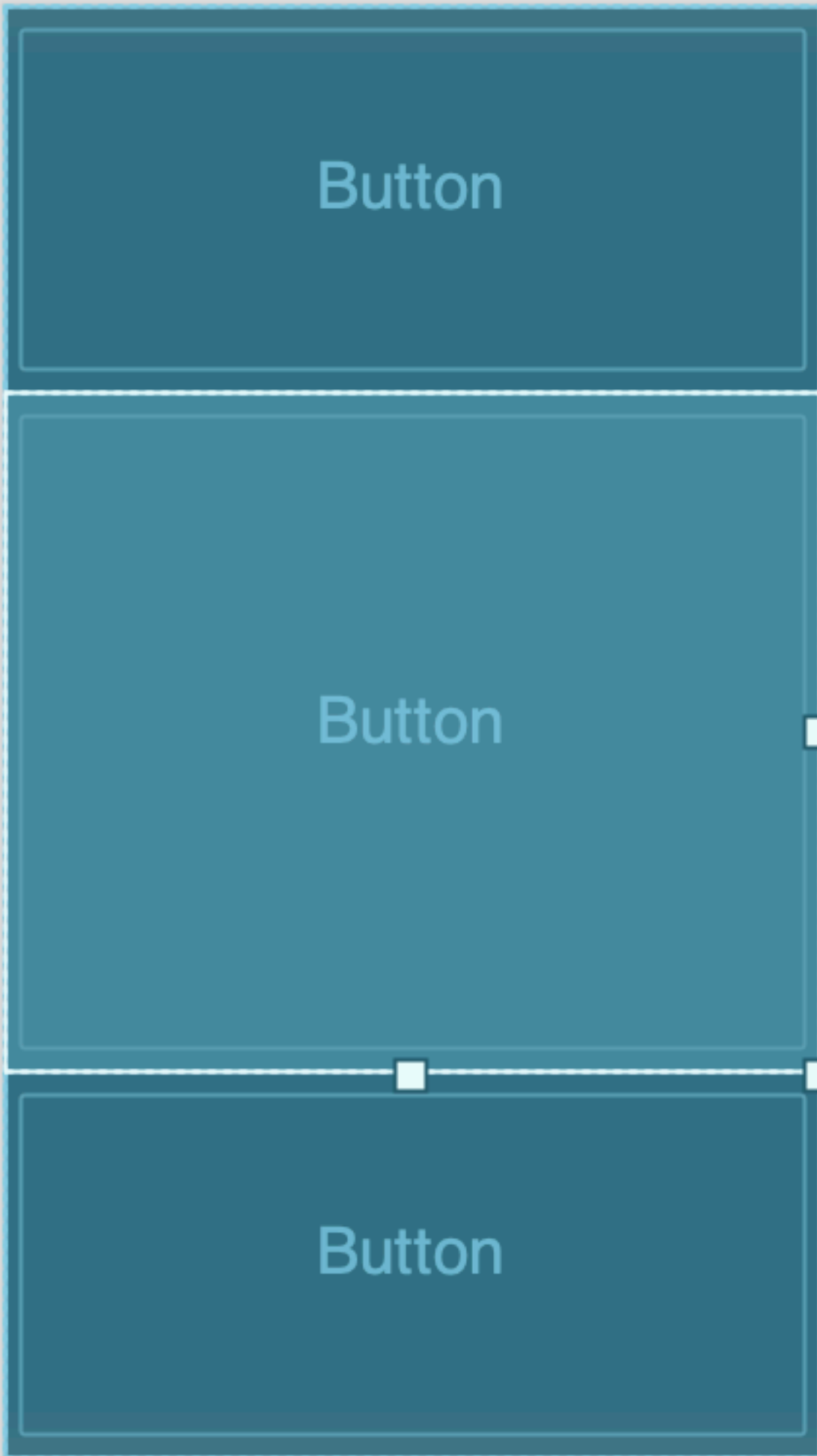
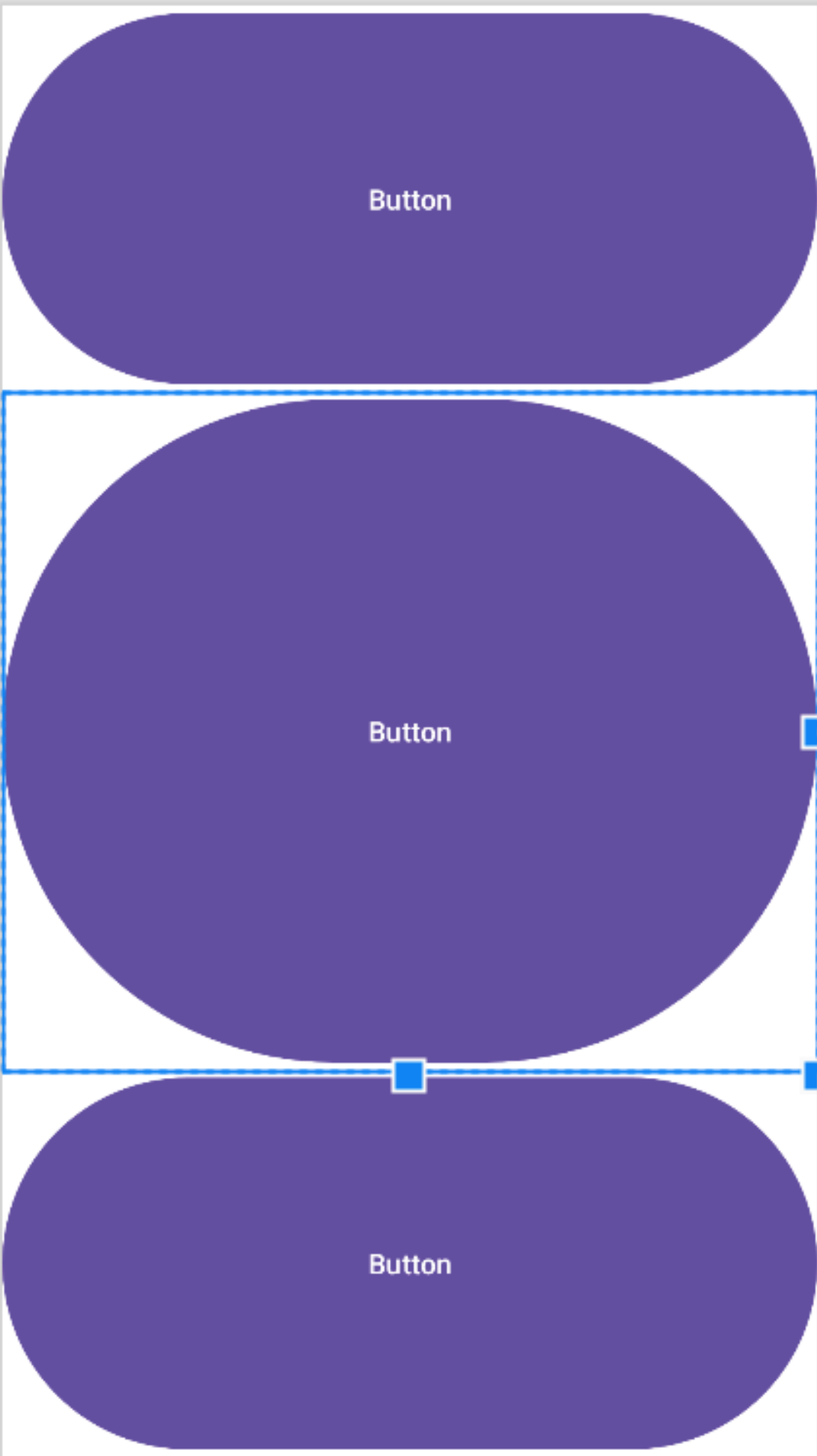
0dp



Component Tree

 Button button2

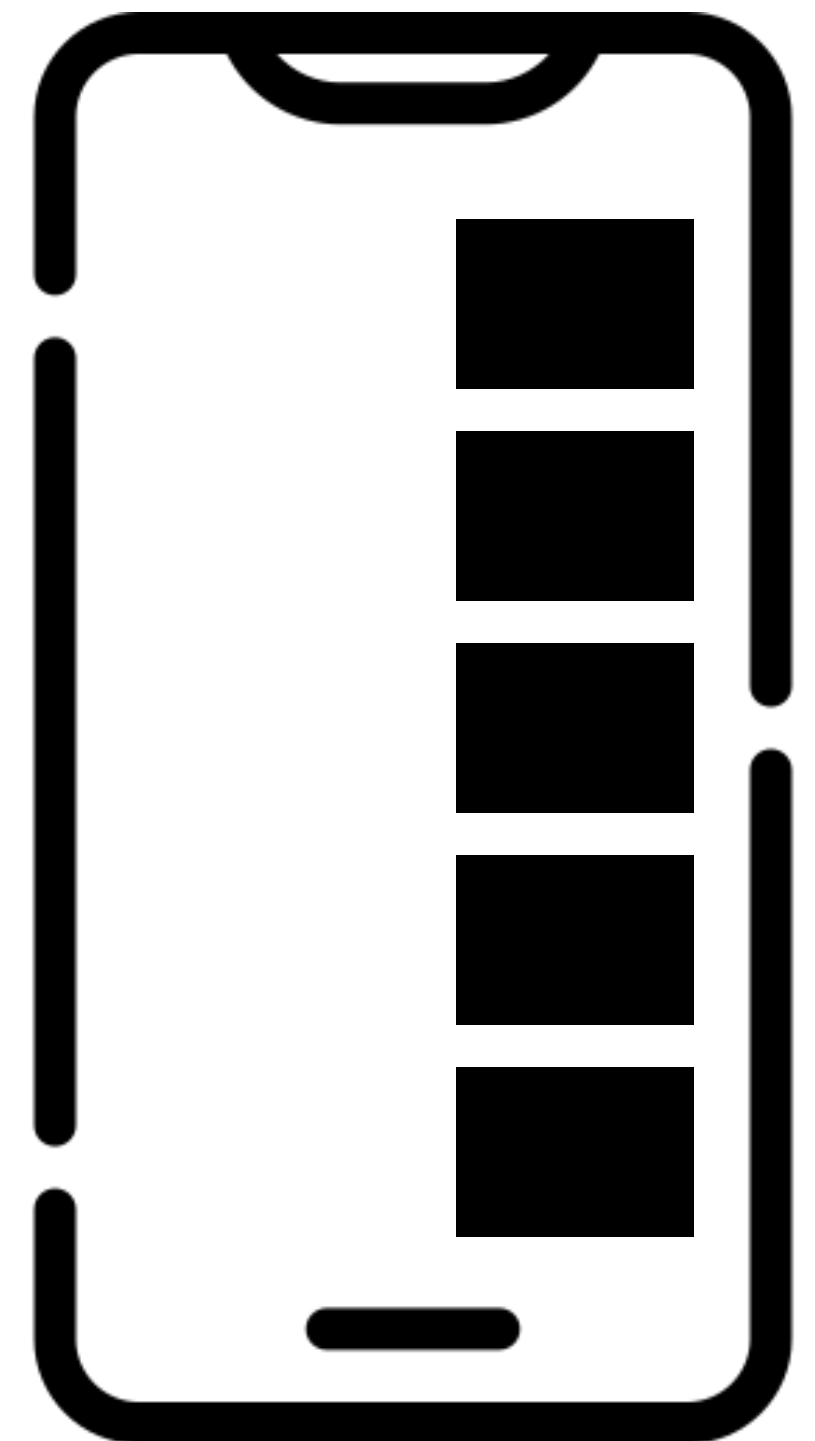
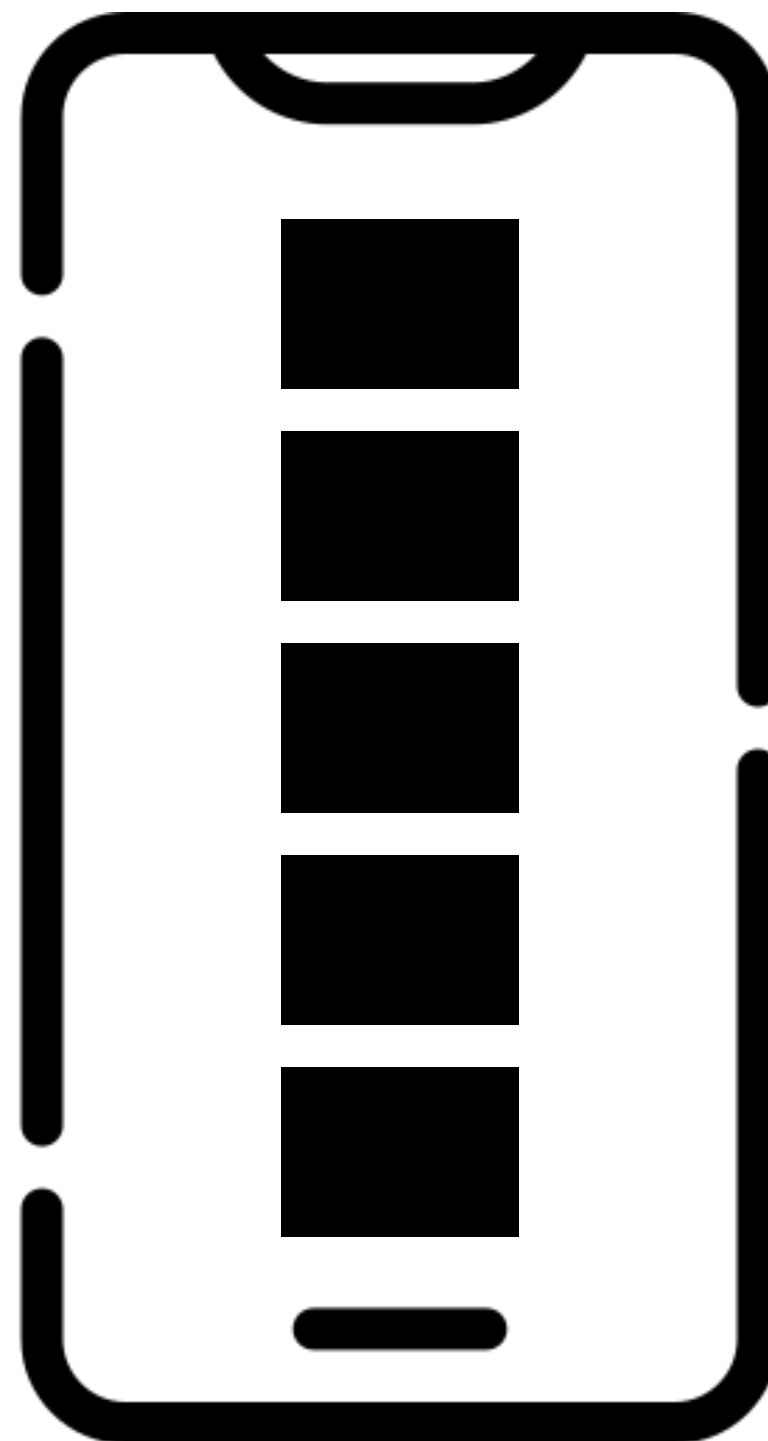
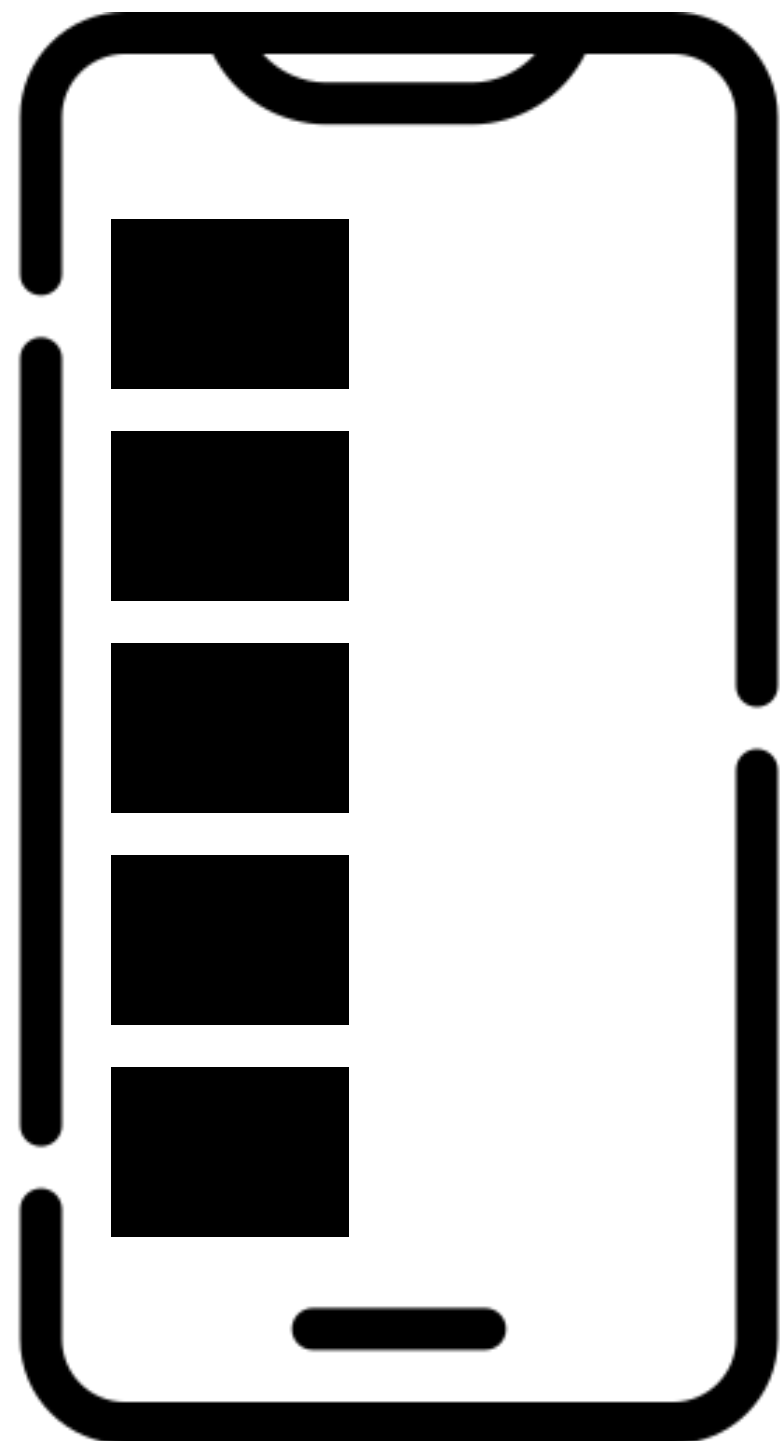
layout_weight	2	
---------------	---	---



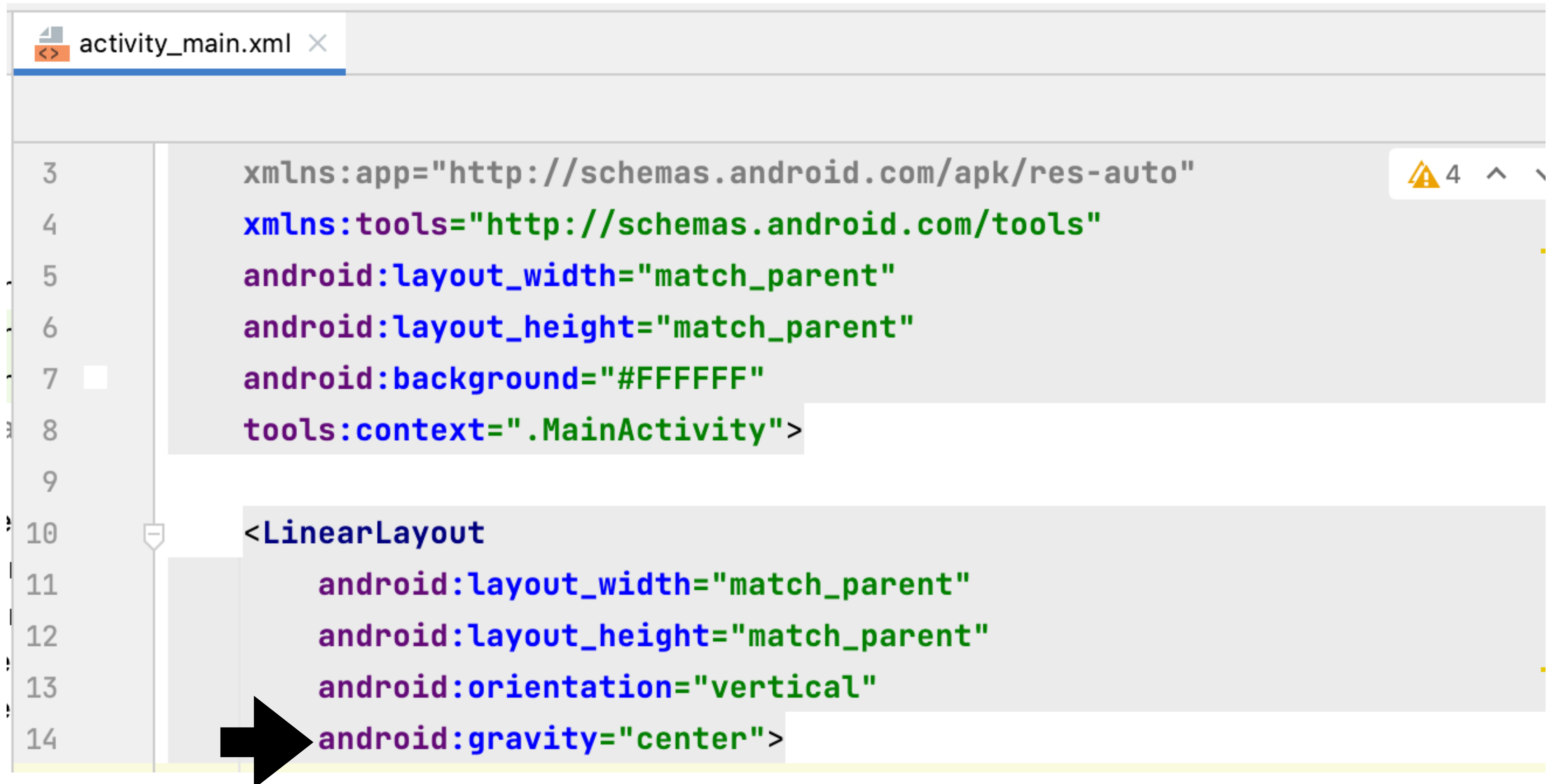
2 LinearLayout

Podemos elegir la alineación de los elementos

`android:gravity`



2 LinearLayout



The screenshot shows the XML editor in Android Studio for the file `activity_main.xml`. The code defines a `LinearLayout` with the following attributes:

```
3  xmlns:app="http://schemas.android.com/apk/res-auto"
4  xmlns:tools="http://schemas.android.com/tools"
5  android:layout_width="match_parent"
6  android:layout_height="match_parent"
7  android:background="#FFFFFF"
8  tools:context=".MainActivity">
9
10 <LinearLayout
11     android:layout_width="match_parent"
12     android:layout_height="match_parent"
13     android:orientation="vertical"
14     android:gravity="center">
```

A large black arrow points to the `android:gravity="center"` attribute on line 14. In the top right corner, there is a warning icon and the number 4.

activity_main.xml

CodeSplitDesign

activity_main.xml

Pixel34MyApplicationDefault (en-us)

0dp

Palette

Common

Text

Buttons

Widgets

Layouts

Containers

Helpers

Google

Legacy

Ab TextView

Button

ImageView

RecyclerView

Fragment...

ScrollView

Switch

Component Tree

ConstraintLayout

LinearLayout (vertical)

button "Button"

button2 "Button"

button3 "Button"

LinearLayout

<unnamed>

> foregroundGravity

gravity

center

Button

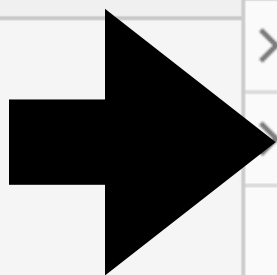
Button

Button

Button

Button

Button

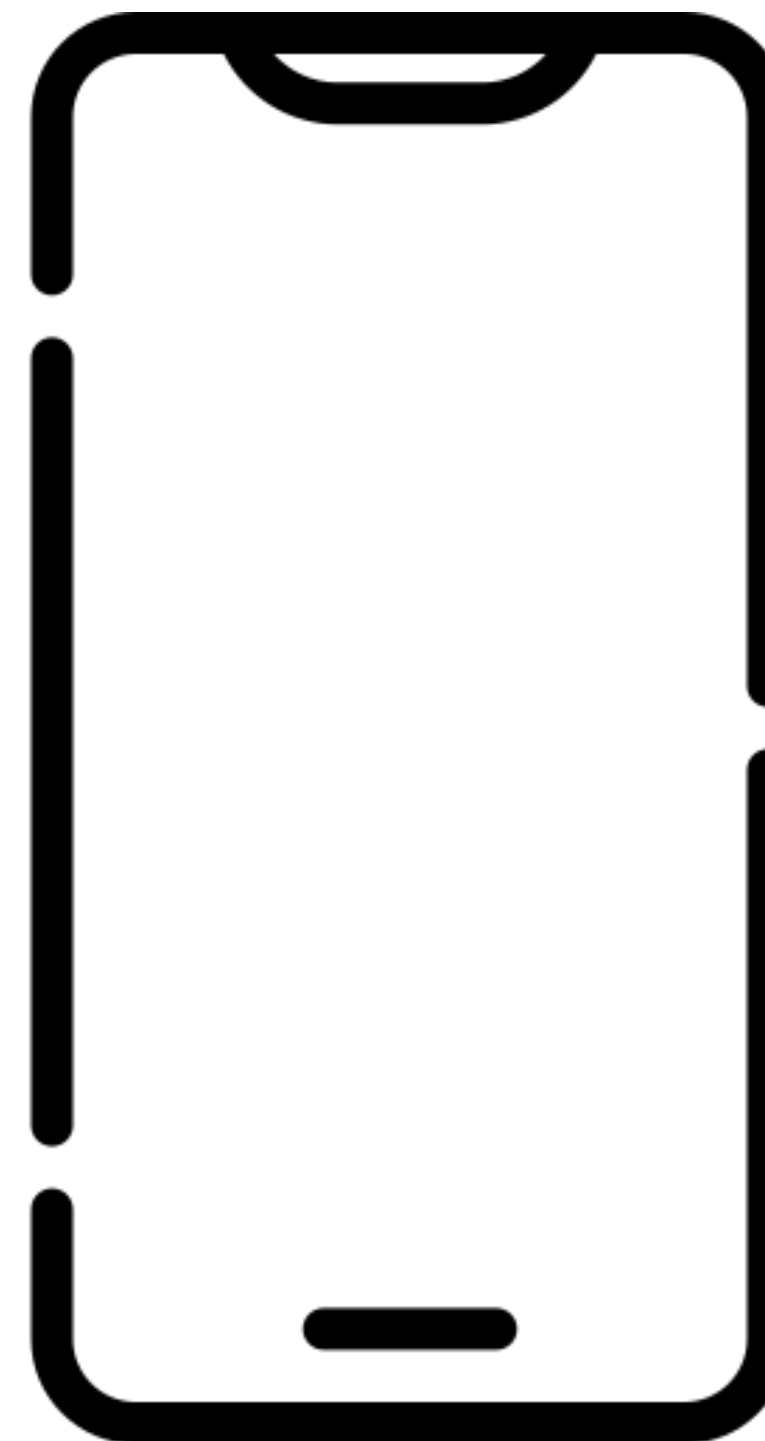
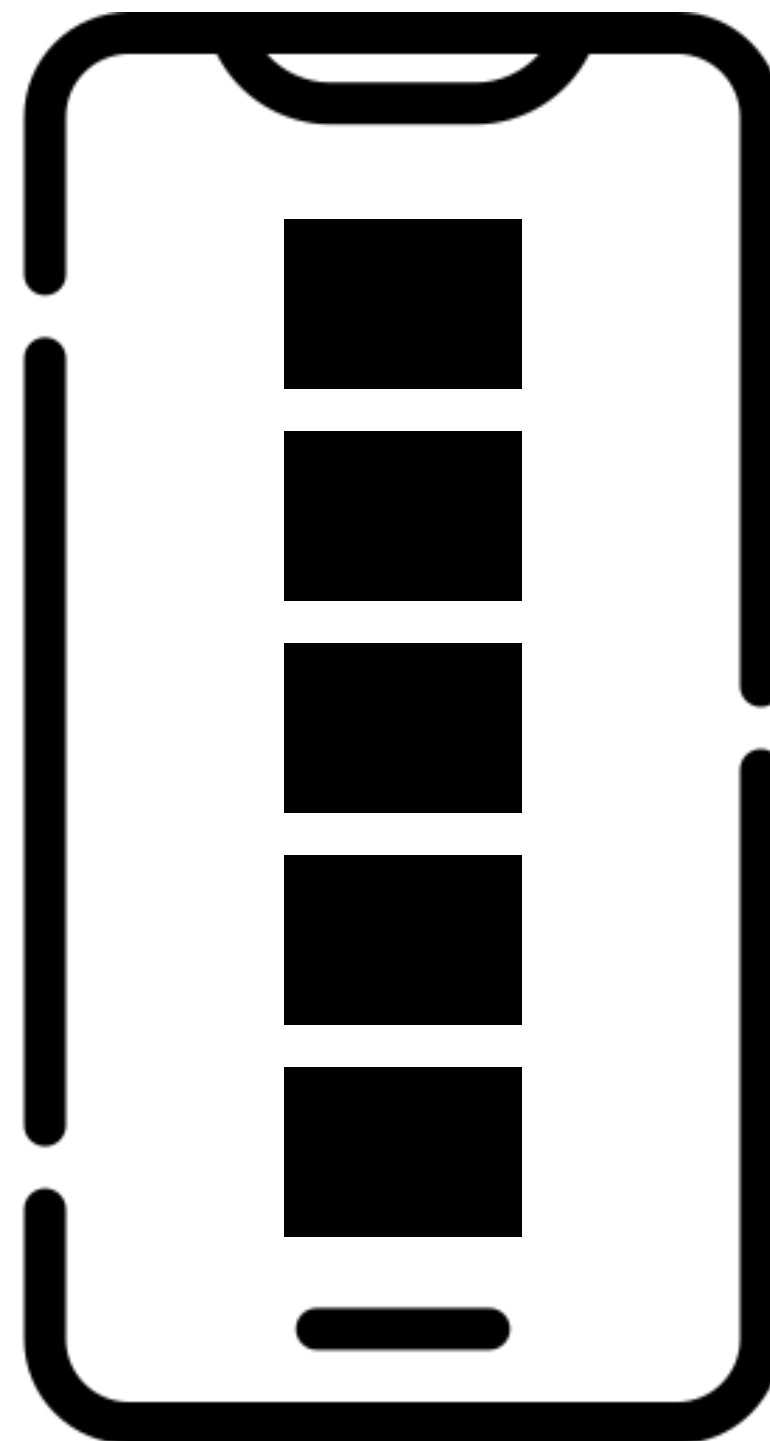


- **Center:** Centra los elementos dentro del Layout tanto de forma horizontal como vertical.
- **Top:** alinea arriba los elementos dentro del Layout.
- **Bottom:** Alinea abajo los elementos dentro del Layout.
- **Right:** Alinea a la derecha los elementos dentro del Layout.
- **Left:** Alinea a la izquierda los elementos dentro del Layout.
- **Center_horizontal:** Alinea al centro de forma horizontal los elementos dentro del Layout.
- **Center_vertical:** Alinea al centro de forma vertical los elementos dentro del Layout.

2 LinearLayout

Podemos ocultar y mostrar todos los elementos

`Android:visibility`

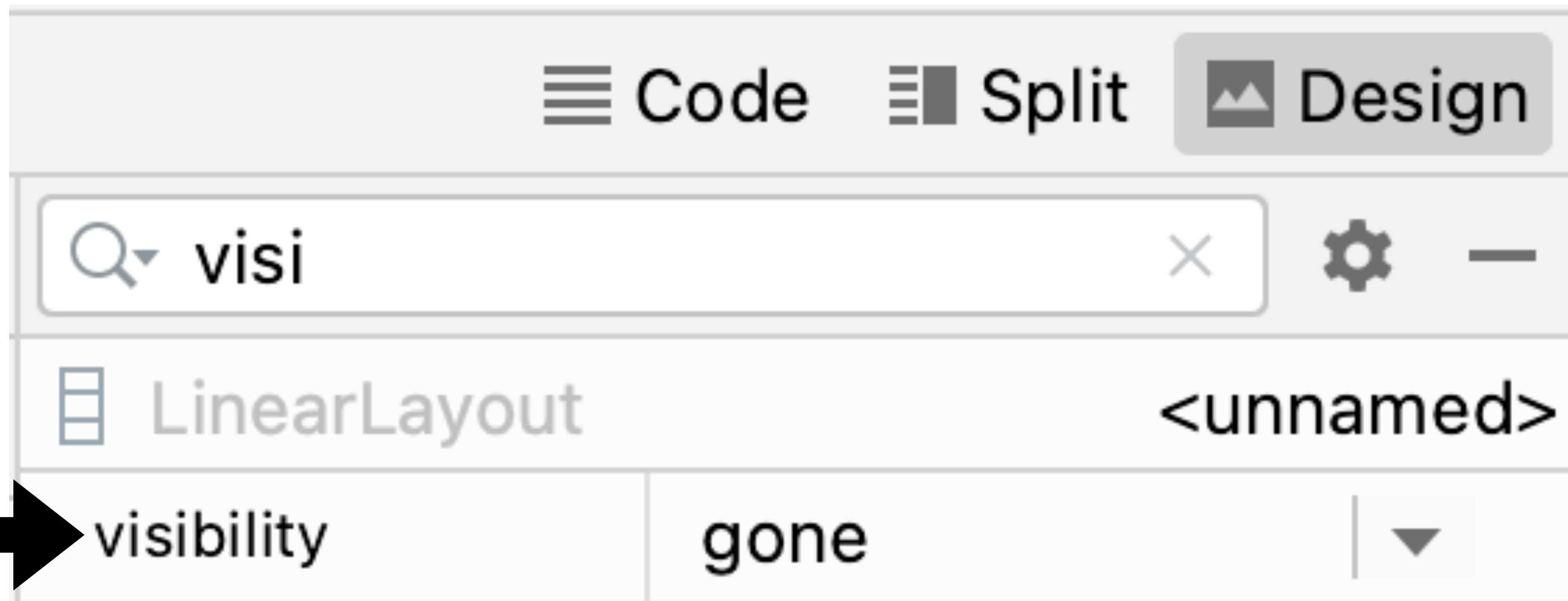


2 LinearLayout

activity_main.xml

```
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   android:background="#FFFFFF"
8   tools:context=".MainActivity">
9
10  <LinearLayout
11    android:layout_width="match_parent"
12    android:layout_height="match_parent"
13    android:gravity="center"
14    android:orientation="vertical"
15    android:visibility="gone">
```

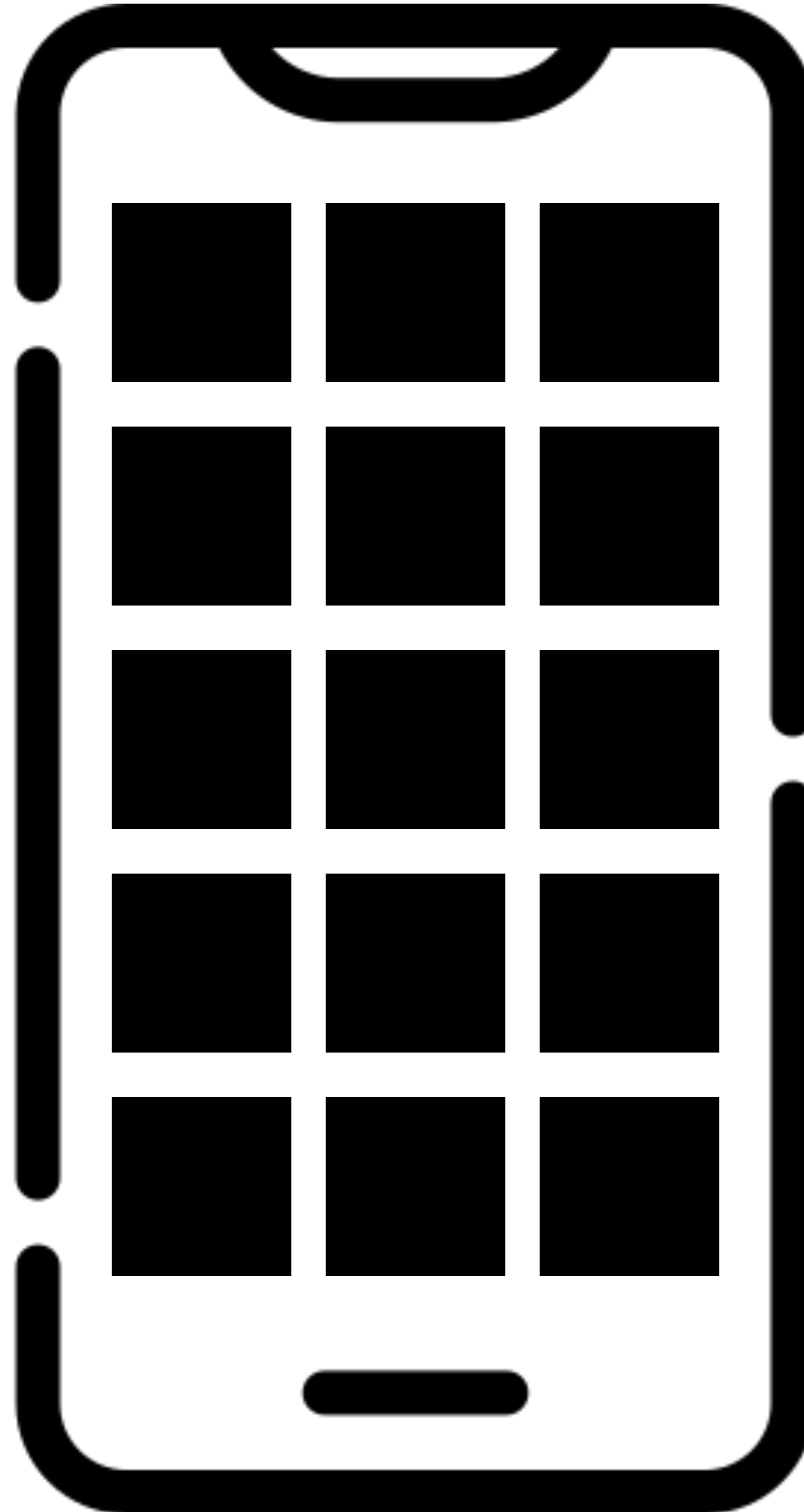
2 LinearLayout



3

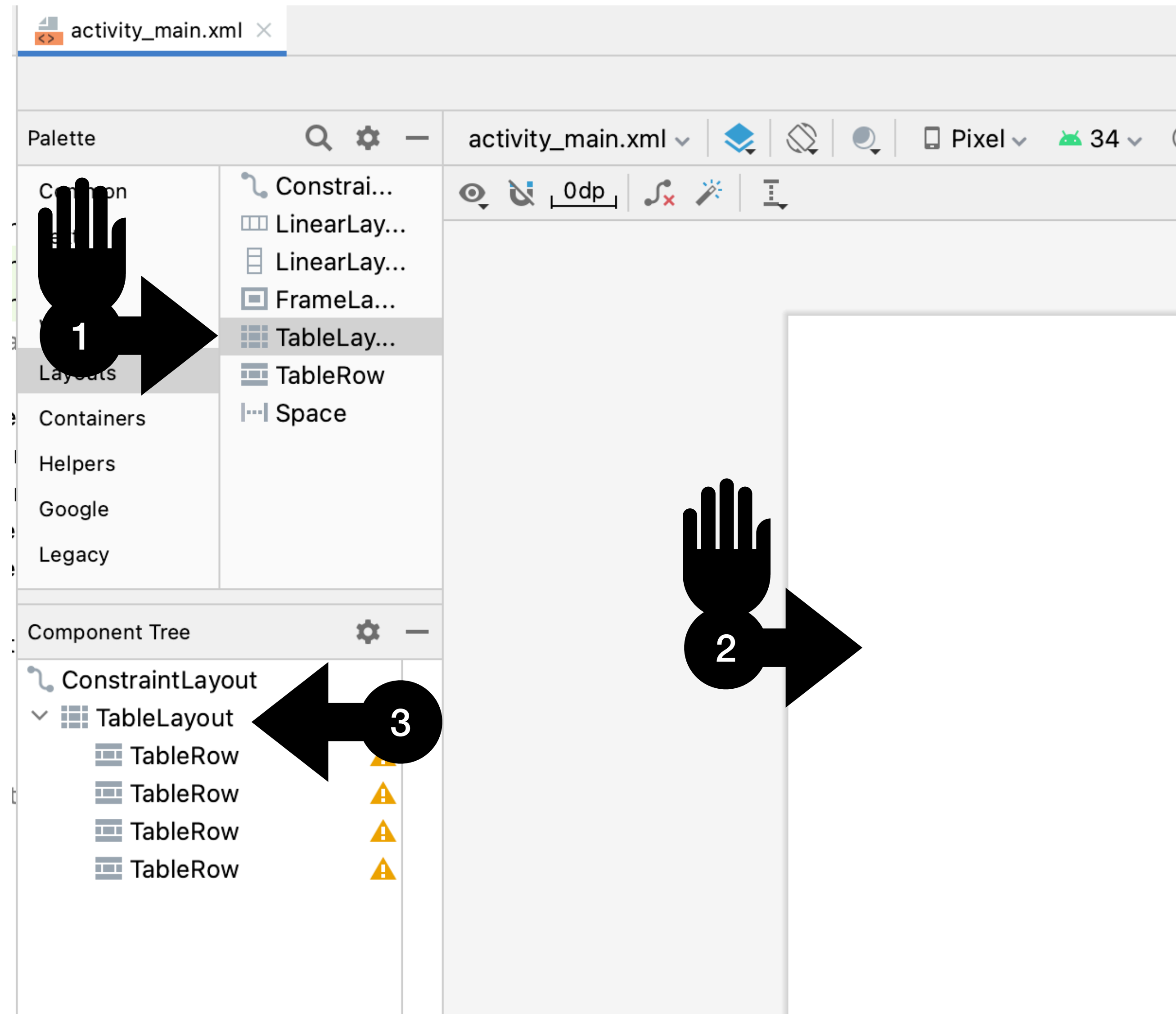
TableLayout

Diseño en forma
de tabla



Podemos definir
el número de
filas y columnas

3 TableLayout



3

TableLayout

Las columnas
se crean
dinámicamente
añadiendo
elementos

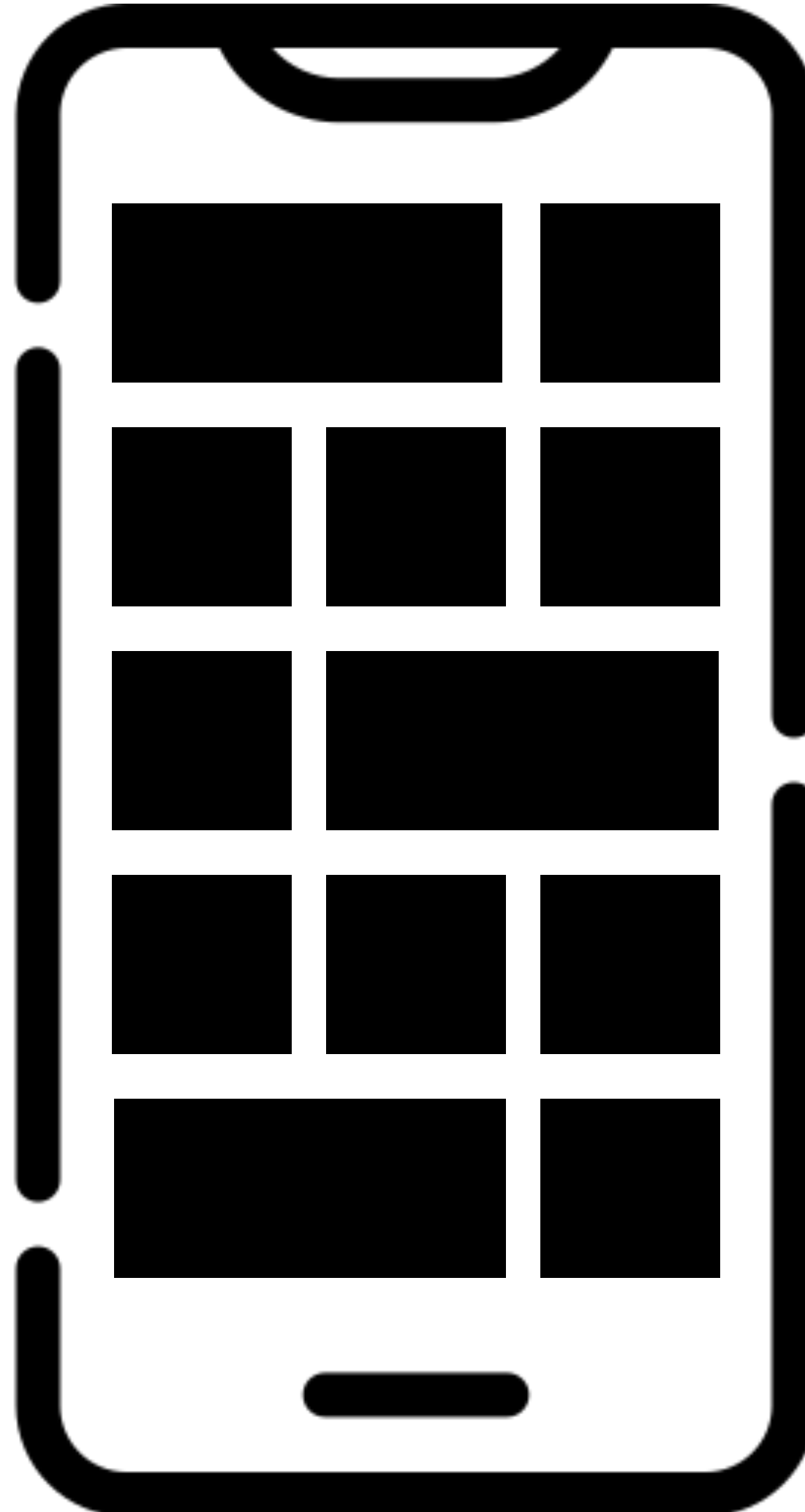
```
11 <TableLayout
12     android:layout_width="match_parent"
13     android:layout_height="match_parent"
14     tools:layout_editor_absoluteX="1dp"
15     tools:layout_editor_absoluteY="1dp">
16     <TableRow
17         android:layout_width="match_parent"
18         android:layout_height="match_parent" />
19
20     <TableRow
21         android:layout_width="match_parent"
22         android:layout_height="match_parent" />
23
24     <TableRow
25         android:layout_width="match_parent"
26         android:layout_height="match_parent" />
27
28     <TableRow
29         android:layout_width="match_parent"
30         android:layout_height="match_parent" />
31 </TableLayout>
```

Número de
columnas = La
que más
columnas tenga

3

TableLayout

Podemos hacer
que el contenido
de una celda
ocupe varias
celdas



`android:layout_span`

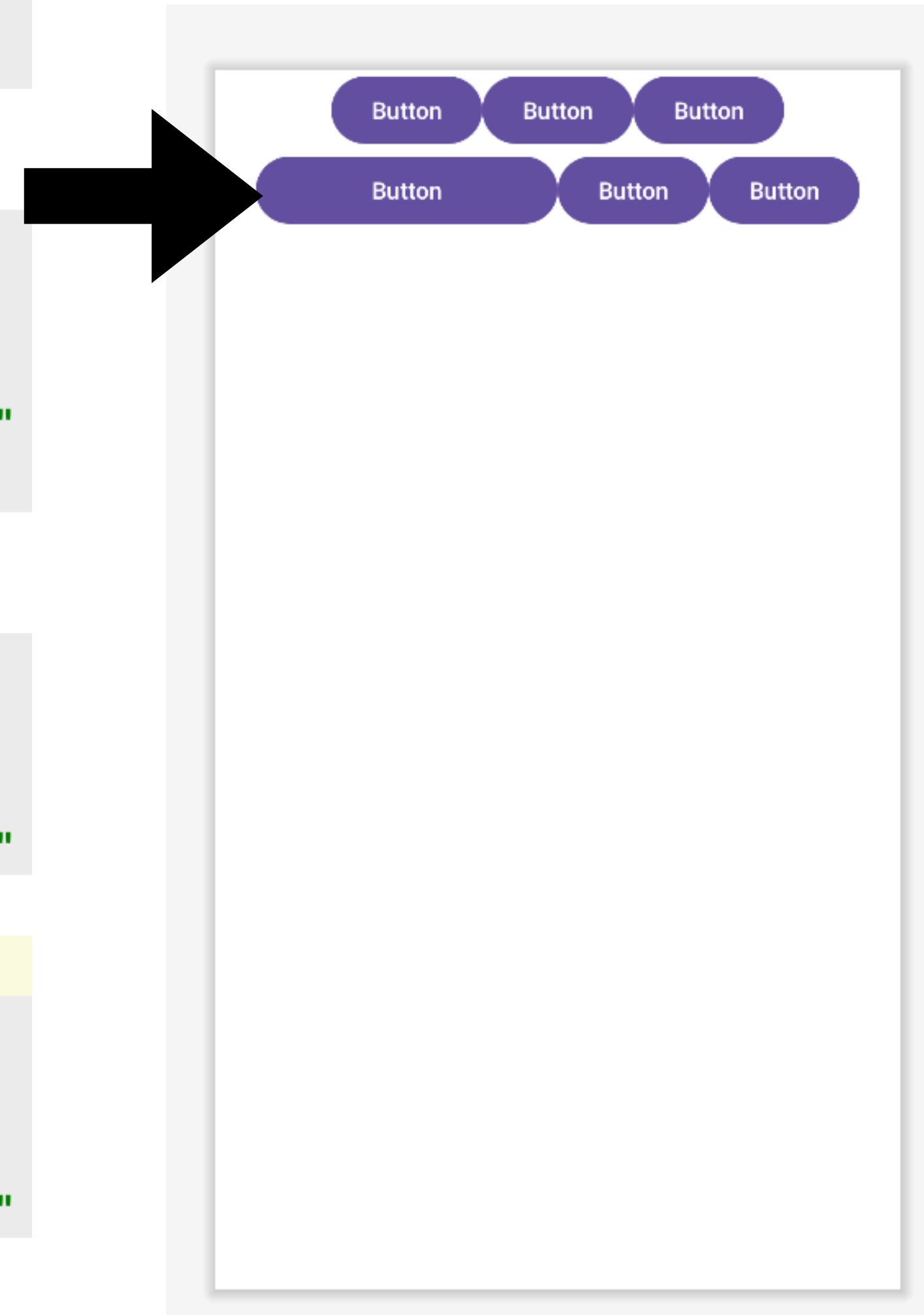
Del elemento
que hemos
añadido

3

TableLayout

Podemos hacer
que el contenido
de una celda
ocupe varias
celdas

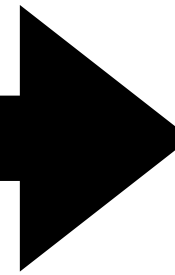
```
<TableRow  
    android:layout_width="match_parent"  
    android:layout_height="match_parent" >  
  
    <Button  
        android:id="@+id/button11"  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_span="2"  
        android:text="Button" />  
  
    <Button  
        android:id="@+id/button12"  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:text="Button" />  
  
    <Button  
        android:id="@+id/button13"  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:text="Button" />  
  
</TableRow>
```



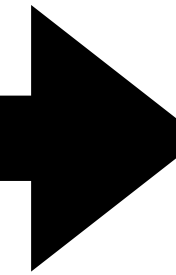
3

TableLayout

El ancho de las
columnas



Mayor columna

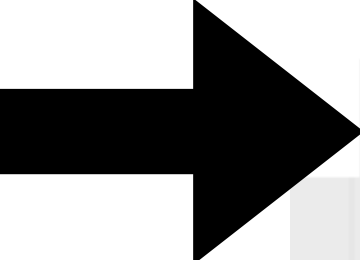


Pero
podemos
cambiarlo

- **android:stretchColumns:** Podremos indicar qué columnas se van a poder expandir para ocupar el espacio libre que han dejado las demás columnas a la derecha.
- **android:shrinkColumns:** Nos permitirá indicar qué columnas se van a poder contraer para dejar espacio, y que se puedan salir por la derecha.
- **android:collapseColumns:** Con esta propiedad vamos a poder indicar qué columnas de nuestra tabla queremos ocultar de forma completa.

3

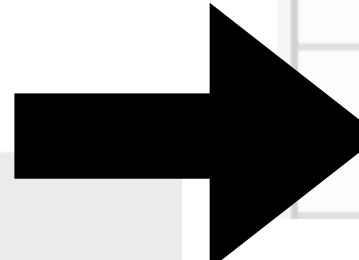
TableLayout



```
<TableLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:layout_editor_absoluteX="1dp"
    tools:layout_editor_absoluteY="1dp"
    android:stretchColumns="0,2">
```

```
<TableRow
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:gravity="center">
```

```
<Button
    android:id="@+id/button8"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Button" />
```

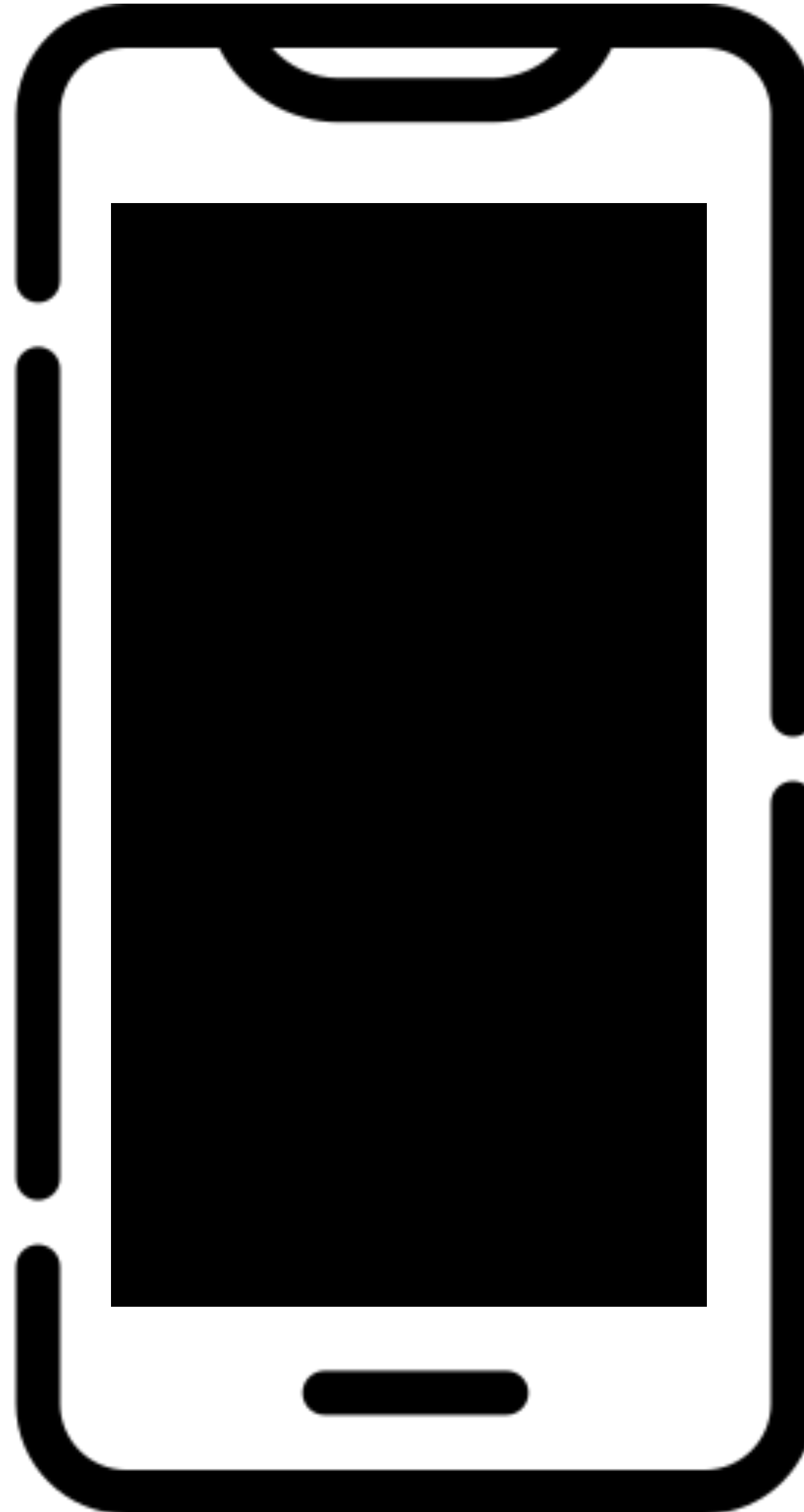


Code Split Design		
stre		
TableLayout <unnamed>		
> layout_constraints		
stretchColumns	0,2	

4

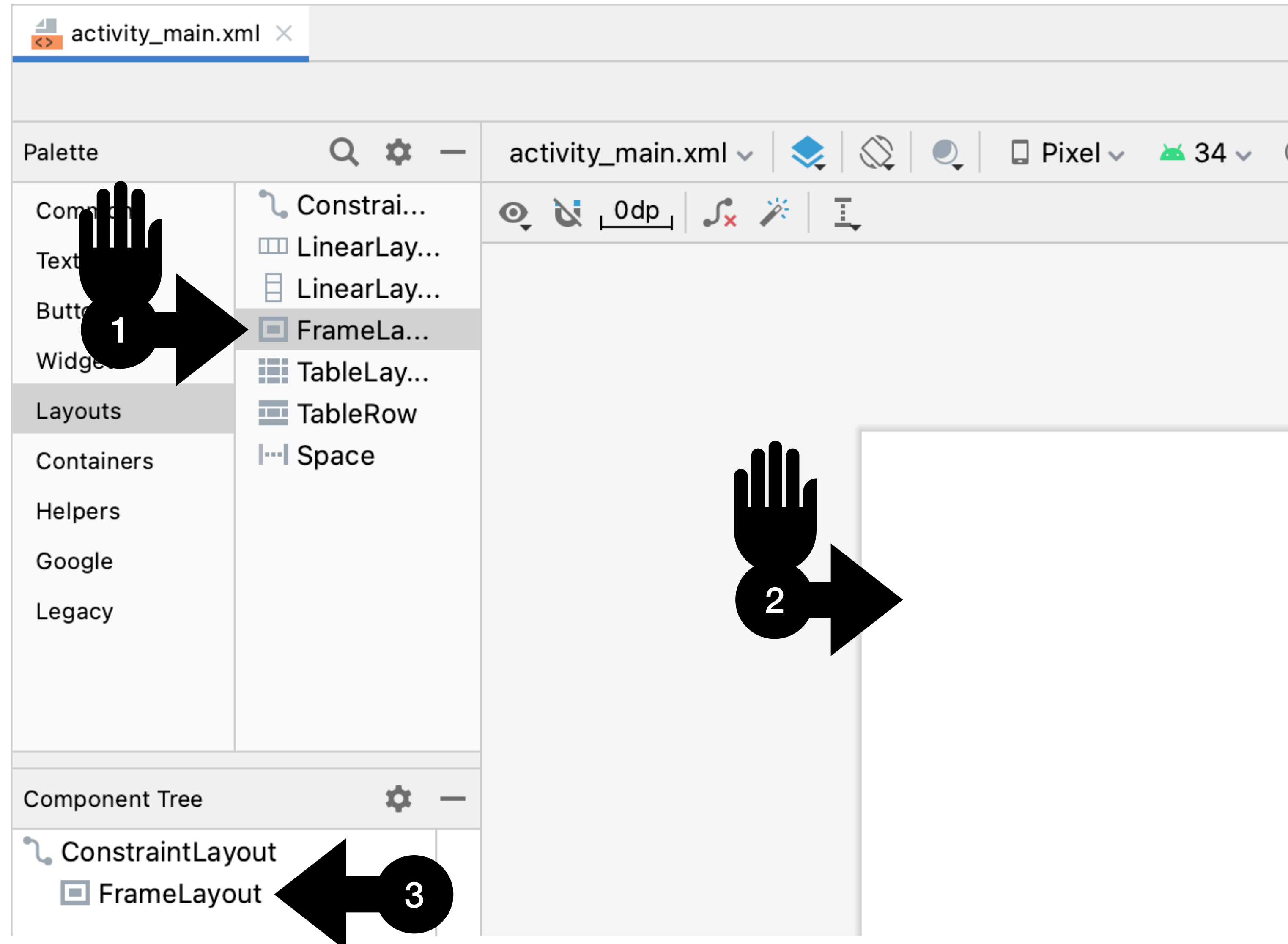
FrameLayout

Es un layout
muy sencillo
que permite
poca
configuración



Se suele utilizar
para colocar un
único elemento
(vista) en su
interior

4 FrameLayout



4 FrameLayout

```
activity_main.xml x
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:background="#FFFFFF"
8     tools:context=".MainActivity">
9
10    <FrameLayout
11        android:layout_width="match_parent"
12        android:layout_height="match_parent"
13        tools:layout_editor_absoluteX="1dp"
14        tools:layout_editor_absoluteY="1dp">
15
16    </FrameLayout>
17 </androidx.constraintlayout.widget.ConstraintLayout>
18
19
```

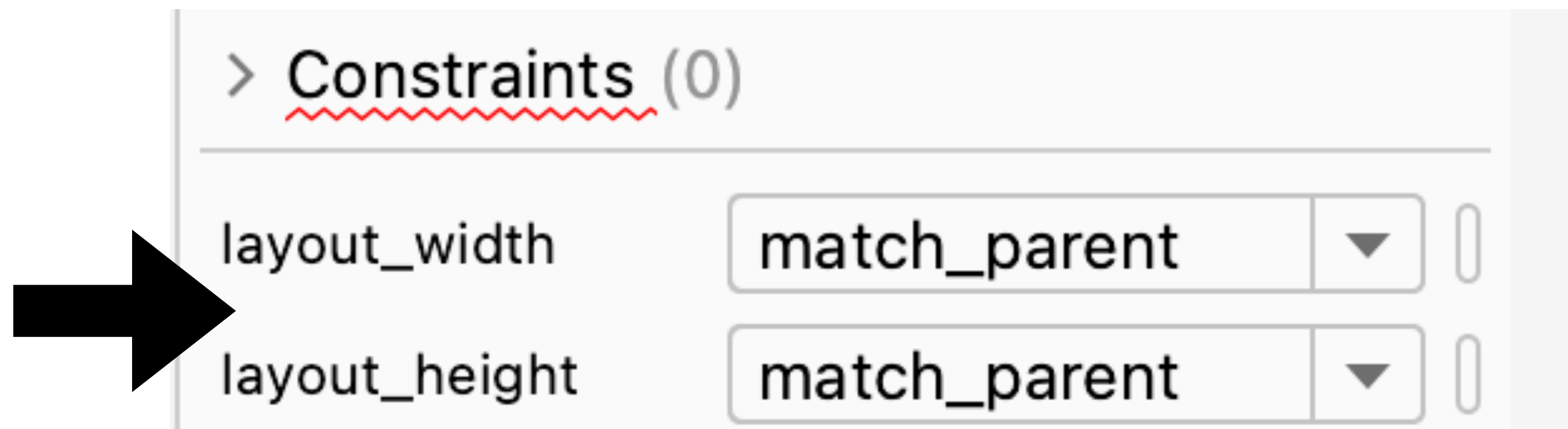
Hay que
indicar el ancho
y el alto

Android:layout_width

Android:layout_height

4 **FrameLayout**

Hay que
indicar el ancho
y el alto



Android:layout_width

Android:layout_height

activity_main.xml

1

<?xml version="1.0" encoding="utf-8"?>

2

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"

3

xmlns:app="http://schemas.android.com/apk/res-auto"

4

xmlns:tools="http://schemas.android.com/tools"

5

android:layout_width="match_parent"

6

android:layout_height="match_parent"

7

android:background="#FFFFFF"

8

tools:context=".MainActivity">

9

<FrameLayout

11

android:layout_width="match_parent"

12

android:layout_height="match_parent"

13

tools:layout_editor_absoluteX="1dp"

14

tools:layout_editor_absoluteY="1dp"

15

>

16

<View

18

android:id="@+id/view"

19

android:layout_width="400dp"

20

android:layout_height="400dp"

21

android:background="@color/greenColor"

22

android:layout_gravity="center"

23

/>

24

</FrameLayout>

25

</androidx.constraintlayout.widget.ConstraintLayout>

activity_main.xml

0dp

Pixel

Palette

Component Tree

5

ConstraintLayout

**Es el tipo más
complejo y que
más
configuración
necesita**

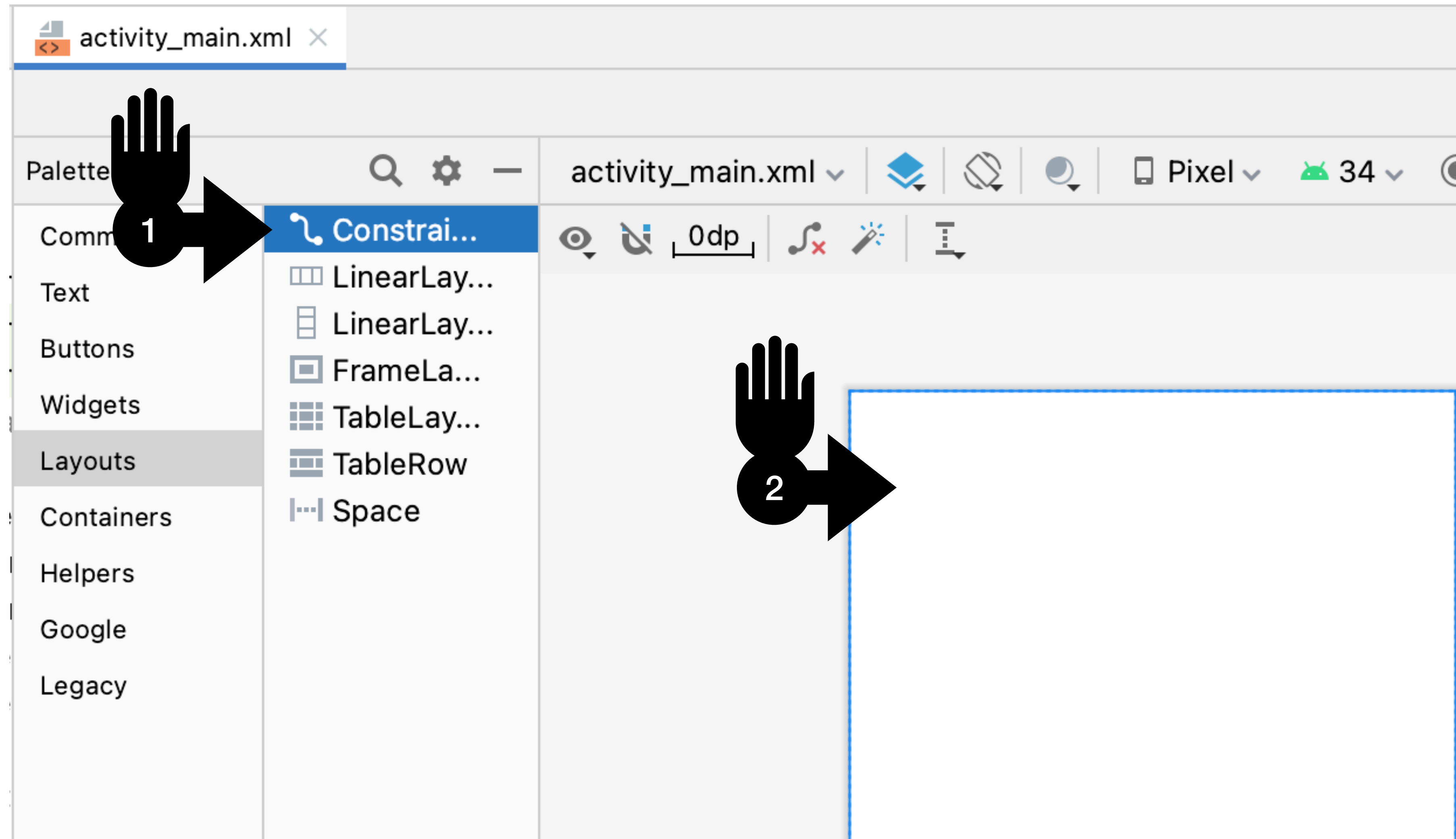
**Nos permite
hacer diseños
mucho más
sofisticados**

**No se
recomienda
tocar el código
generado por su
complejidad**

**Utilizaremos la
interfaz de
diseño de
Android**

5

ConstraintLayout



5

ConstraintLayout

**Quando
colocamos un
elemento en
este layout**

```
graph LR; A[Quando colocamos un elemento en este layout] --> B[Definir restricción vertical]; A --> C[Definir restricción horizontal];
```

**Definir
restricción
vertical**

**Definir
restricción
horizontal**

5

ConstraintLayout

activity_main.xml | Pixel | 34 | MyApplication | Default (en-us)

0dp

Button

Button

Attributes

Button button3

id button3

> Declared Attributes + -

> Layout

Constraint Widget

0 0

> Constraints (2)

layout_width wrap_content

layout_height wrap_content

visibility

visibility

> Transforms

> Common Attributes

> All Attributes

6

Pestañas

No lo vamos a ver por
ahora