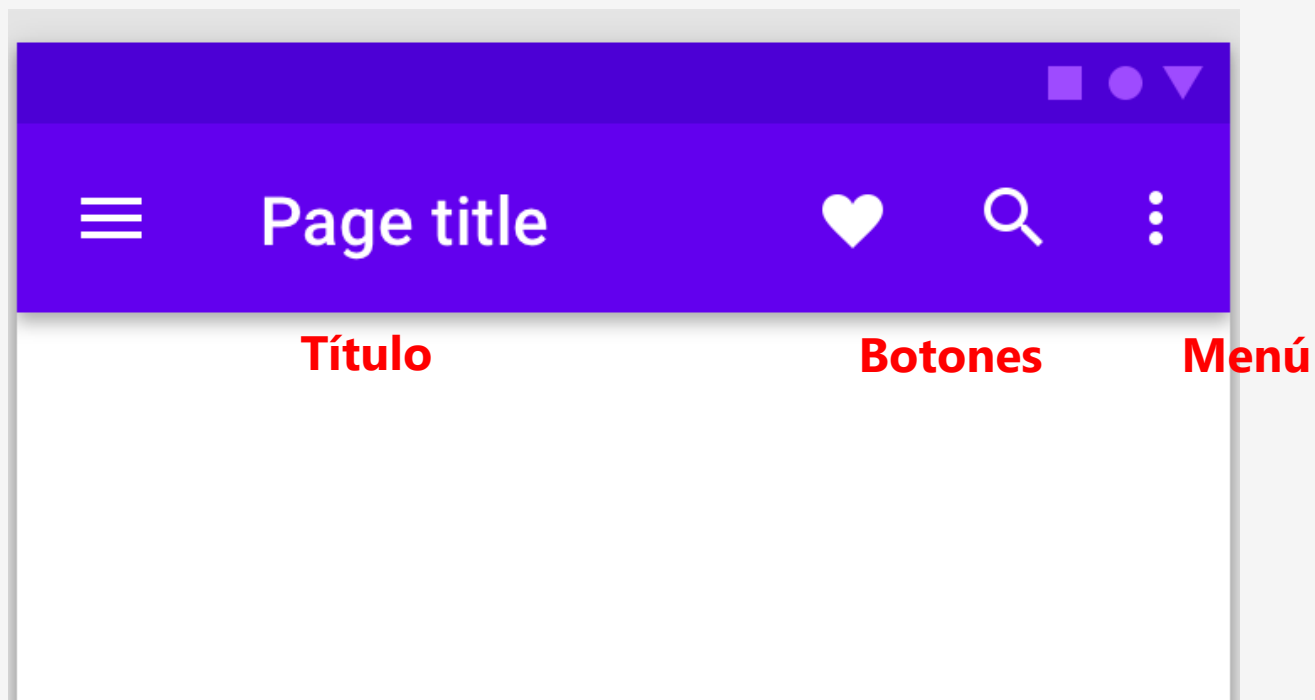


Componentes de Navegación

Clase Nro. 7

Teófilo Copa F.

Barra de la App(ActionBar)

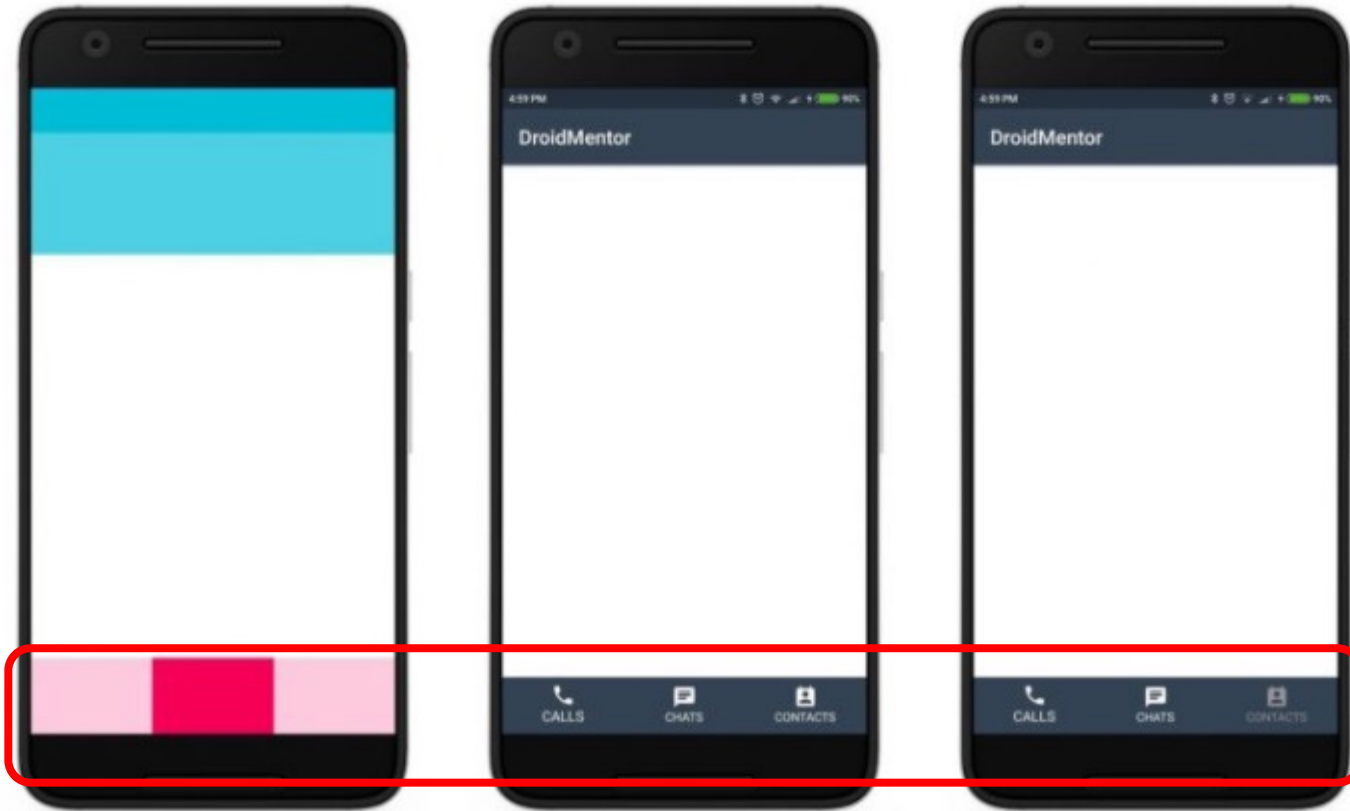


- Es la barra de título y herramientas que aparece en la parte superior.
- Esta compuesto generalmente de un conjunto de botones y un menú desplegable.

Pasos para crear un Action Bar

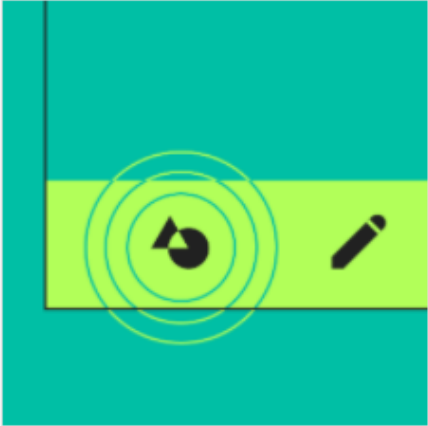
1. **Crear un proyecto** (File -> New Project) con una actividad vacía (Empty Activity).
2. **Crear un nuevo *resource*** de tipo layout (*File->New->Android Resource File*) que contiene un toolbar.
3. **Crear un nuevo resource de tipo menú** (File->New->Menu Resource File)
4. **Agregar El toolbar y el menú a cada** actividad en donde deseemos que se muestre el ActionBar.

Bottom Navigation (Navegación Inferior)



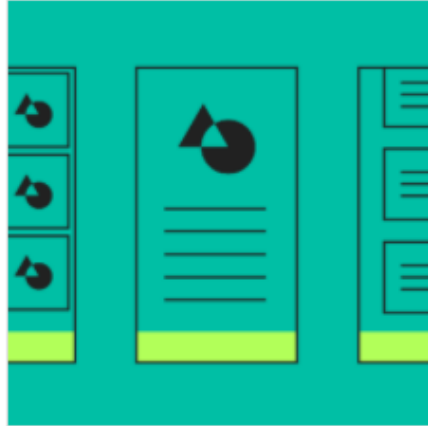
Es un componente que permite el **movimiento entre destinos principales** de una app.

Bottom Navigation (Usos)



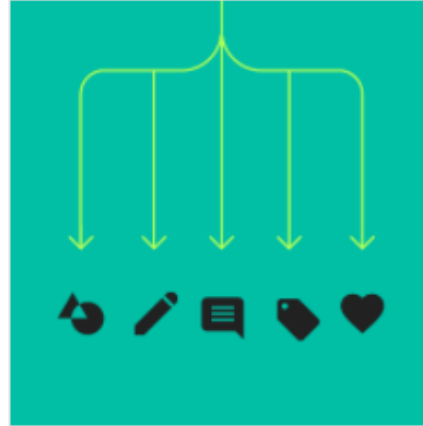
Ergonómico

La barra de navegación inferior es de fácil acceso en un dispositivo móvil portátil.



Consistente

Cuando se utiliza, la barra de navegación inferior aparece en la parte inferior de cada pantalla.

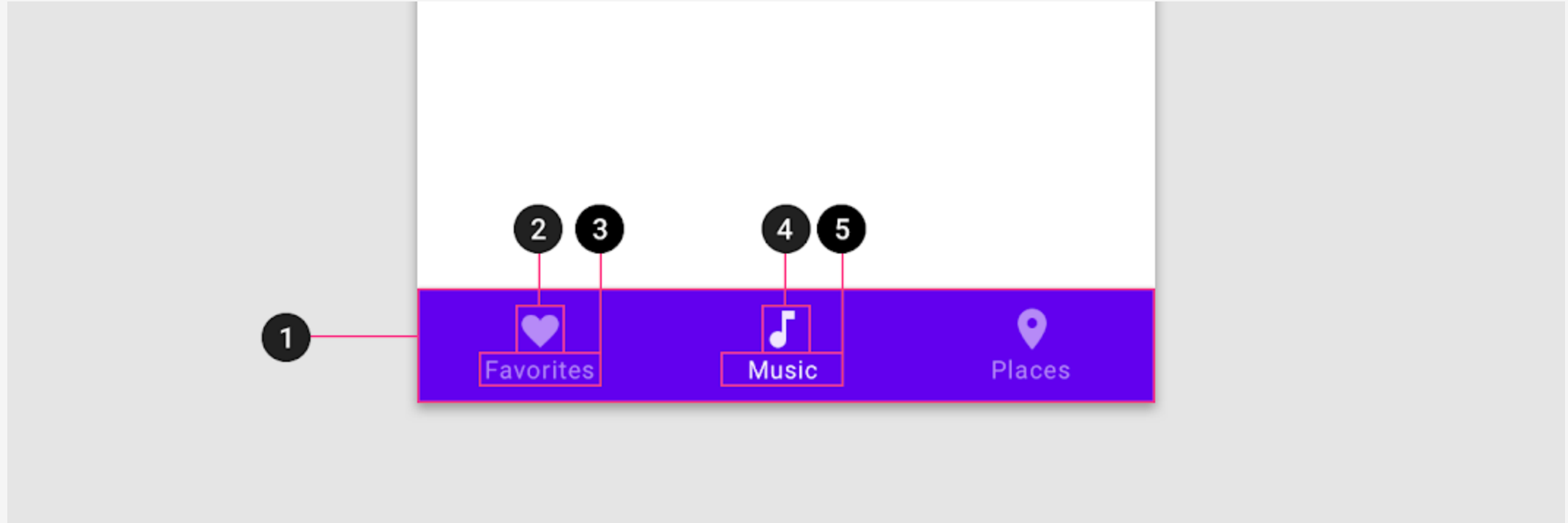


relacionado

Los destinos de la barra de navegación inferior deben ser de igual importancia.

Generalmente se usa cuando se tiene de **3 a 5 destinos principales.**

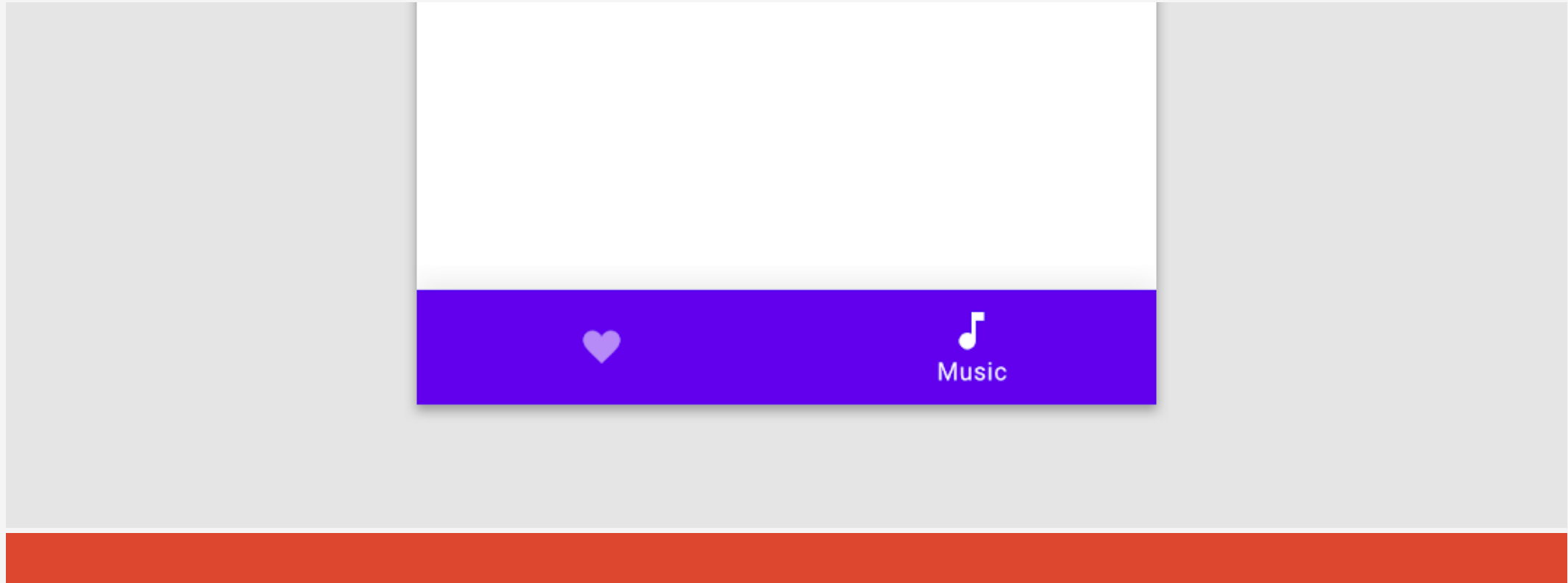
Bottom Navigation (Anatomía)



1. Contenedor
2. Icono inactivo
3. Etiqueta de texto inactivo

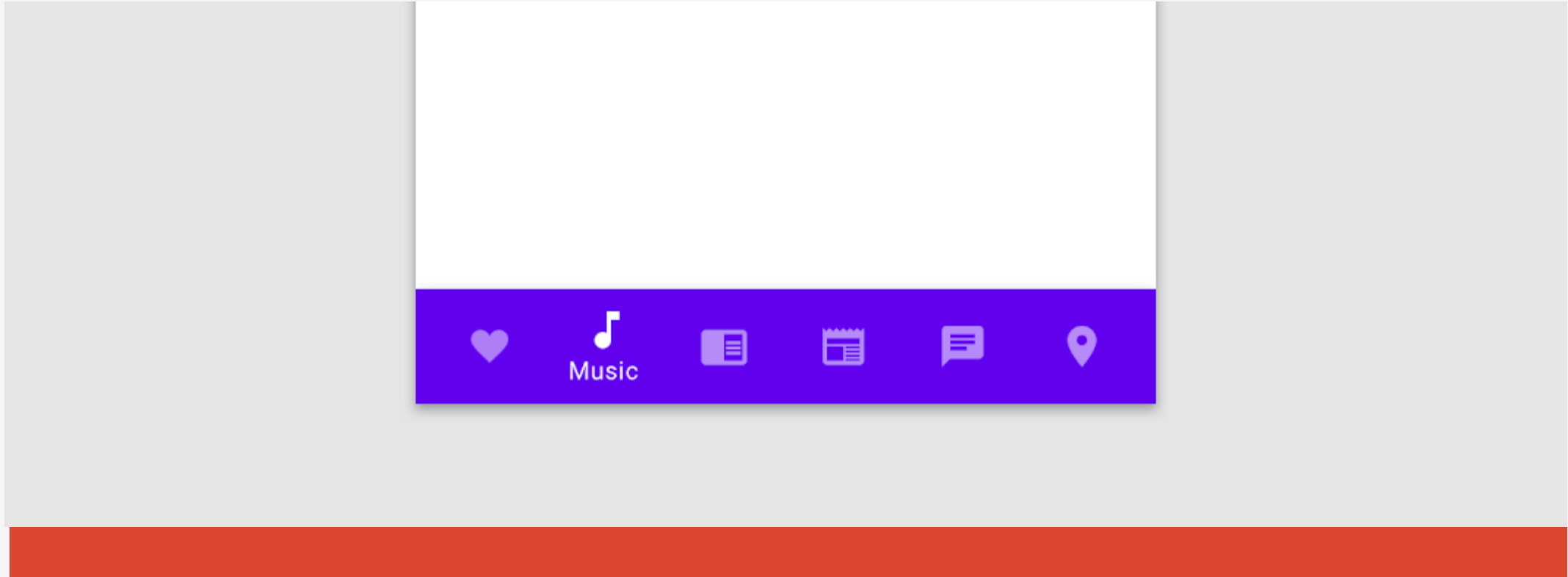
4. Icono activo
5. Etiqueta de texto activa

Bottom Navigation (NO se usa para)



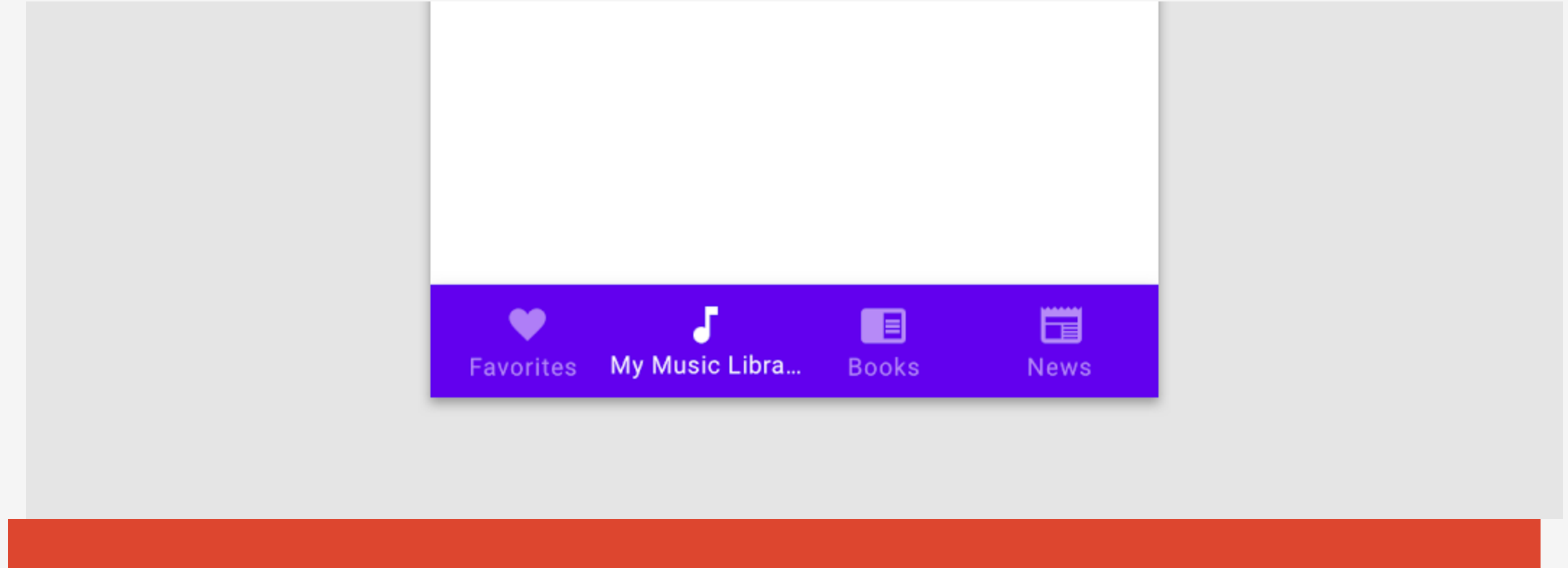
NO utilice una barra de navegación inferior para menos de tres destinos (use pestañas en su lugar)

Bottom Navigation (NO se usa para)



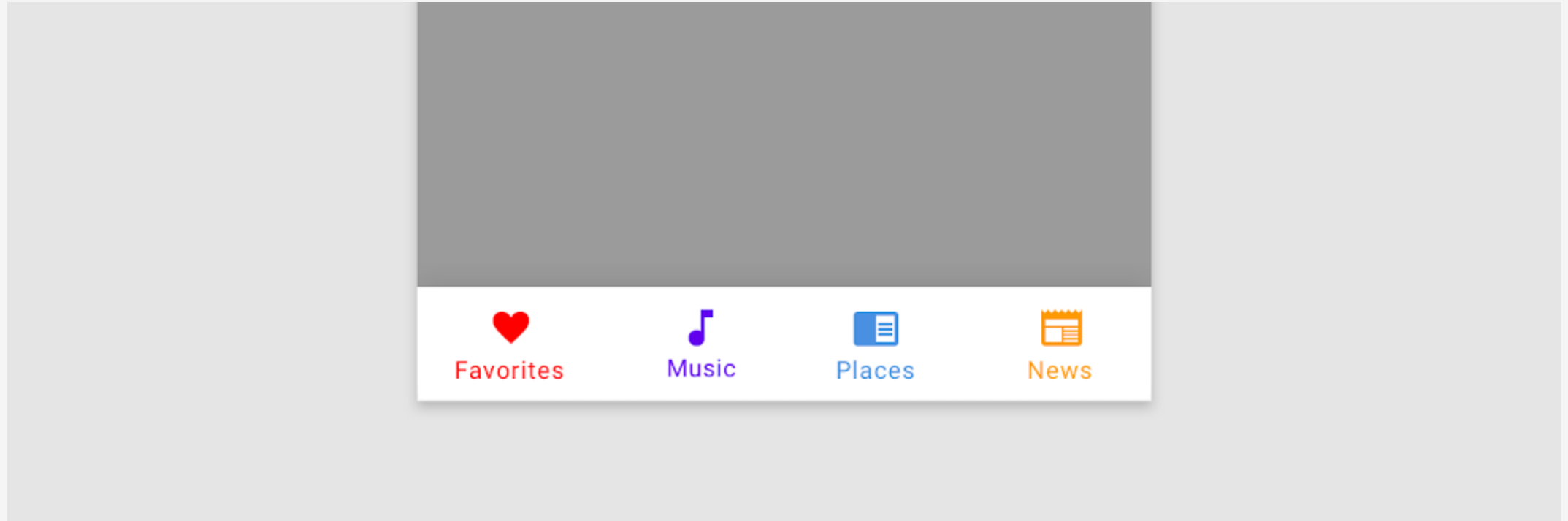
NO uses más de cinco destinos. Para esos casos, pruebe pestañas o un cajón de navegación.

Bottom Navigation (NO se usa para)



NO truncar el texto de un ítem del menú

Bottom Navigation (NO se usa para)



NO No utilice varios o bajos colores de contraste en una barra de navegación inferior.

Pasos para crear una navegación inferior

- 1. Adicionar** el *bottomNavigationView* al **layout principal** (FrameLayout).
- 2. Crear** los **elementos de menú** para la barra de navegación inferior.
- 3. Inicializar** el componente *bottomNavigationView*.
4. Creación de los **fragmentos** que aparecerán al hacer clic.
5. Configuración de **eventos** clic de la navegación.

Pasos para crear una navegación inferior

```
1 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
2     xmlns:app="http://schemas.android.com/apk/res-auto"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     tools:context=".MainActivity"
7     >
8     <FrameLayout
9         android:layout_width="match_parent"
10        android:layout_height="match_parent"
11        android:id="@+id/contenedorPrincipal"
12    >
13        <com.google.android.material.bottomnavigation.BottomNavigationView
14            android:id="@+id/navBottom"
15            android:layout_width="match_parent"
16            android:layout_height="wrap_content"
17            android:layout_gravity="start"
18            android:background="@color/purple_700"
19            app:itemIconTint="#FFFFFF"
20            app:itemTextColor="@color/white"
21            app:menu="@menu/menu_bottom_navigation" />
22    </FrameLayout>
23 </androidx.constraintlayout.widget.ConstraintLayout>
```

Pasos para crear una navegación inferior

```
menu_bottom_navigation.xml X
2  <menu xmlns:android="http://schemas.android.com/apk/res/android">
3
4      <item
5          android:id="@+id/itemInicio"
6          android:title="Inicio"
7
8          android:icon="@drawable/ic_home"
9      />
10     <item
11         android:id="@+id/itemBuscar"
12         android:title="Buscar"
13         android:icon="@drawable/ic_buscar"
14     />
15     <item
16         android:id="@+id/itemBiblioteca"
17         android:title="Biblioteca"
18         android:icon="@drawable/ic_biblioteca"/>
19 </menu>
```

Pasos para crear una navegación inferior

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    fragmentInicio = new FragmentInicio();  
    fragmentBuscar = new FragmentBuscar();  
    fragmentBiblioteca = new FragmentBiblioteca();  
  
    navBottom = findViewById(R.id.navBottom);  
    navBottom.setNavigationItemSelectedListener(this);  
}
```

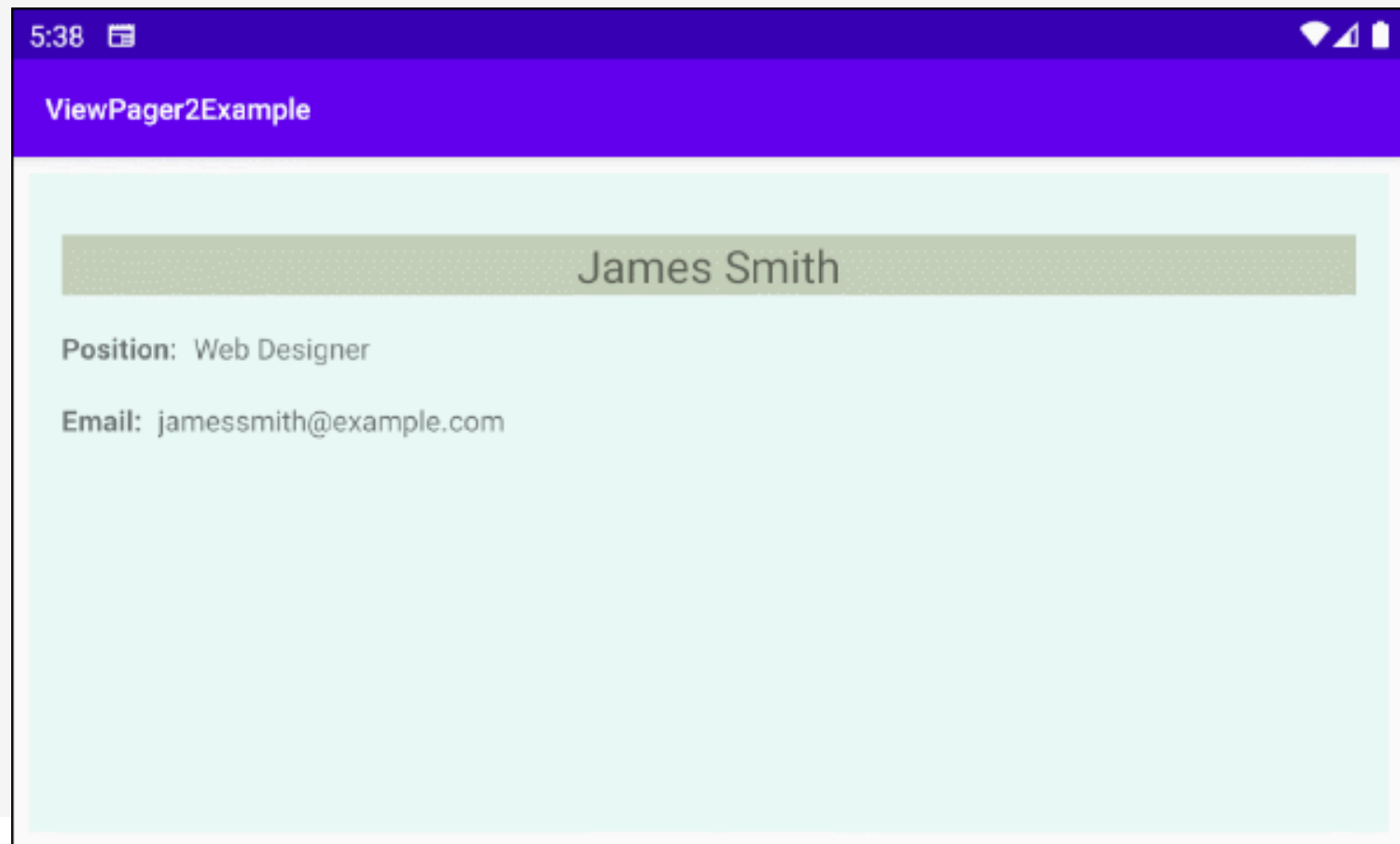
Pasos para crear una navegación inferior

```
@Override
public boolean onNavigationItemSelected(@NonNull MenuItem item) {

    int id = item.getItemId();
    if(id == R.id.itemBuscar){
        abrirFragment(fragmentBuscar);
        return true;
    }else if( id == R.id.itemBiblioteca){
        abrirFragment(fragmentBiblioteca);
        return true;
    }else{
        abrirFragment(fragmentInicio);
        return true;
    }
}
```

Navegación con desplazamientos (ViewPager)

- ***ViewPager*** es el componente que permite mostrar distintas pantallas mediante desplazamientos, como si se tratara de una presentación Powerpoint.
- ***ViewPager2*** es la versión mejorada de *ViewPager*.



Pasos para crear una Navegación con desplazamientos

1. Crear un nuevo proyecto Android
2. Crear los fragmentos que mostrará el ViewPager
3. Implementar un PagerAdapter
4. Asociar el PagerAdapter al ViewPager

Agregar el ViewPager al Layout Principal

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <androidx.viewpager2.widget.ViewPager2
        android:id="@+id/viewPager"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Crear los fragmentos que se mostrará el ViewPager

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#FF8F00"
    tools:context=".FragmentUno"
    tools:visibility="visible">

    <!-- TODO: Update blank fragment layout -->
    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:text="Uno"
        android:textColor="#F8F6F6"
        android:textSize="60sp" />

</FrameLayout>
```

Crear un PagerAdapter

```
class MyViewPagerAdapter extends FragmentStateAdapter {
    public MyViewPagerAdapter(FragmentActivity fa) {
        super(fa);
    }

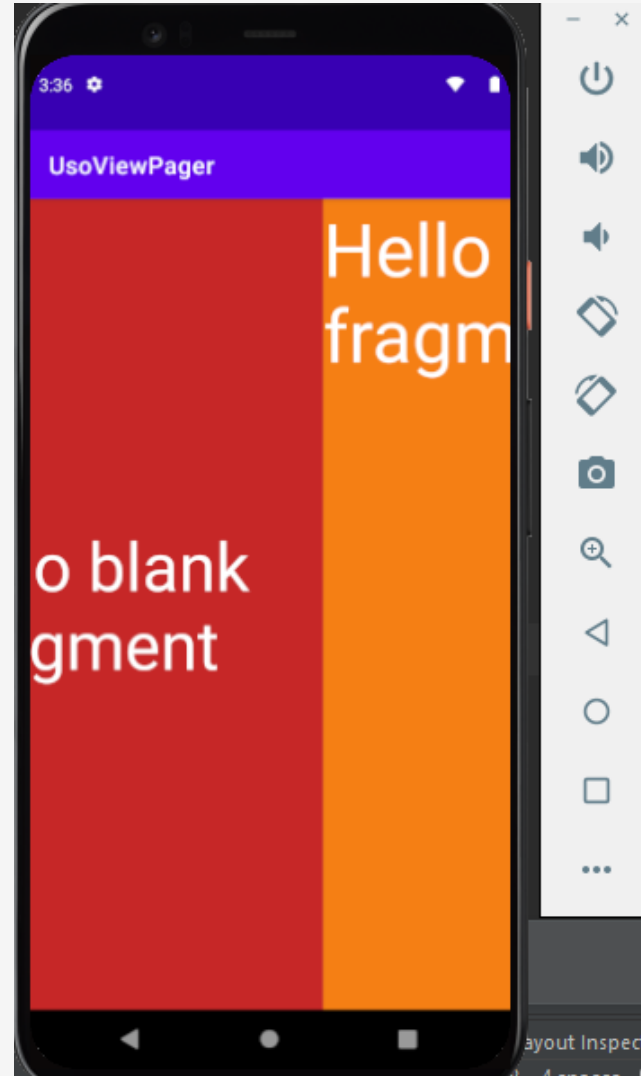
    @Override
    public Fragment createFragment(int position) {
        if(position == 0){
            return new FragmentUno();
        }
        return new FragmentDos();
    }

    @Override
    public int getItemCount() {
        return 2; //indica el numero paginas que se mostrara
    }
}
```

Asociar el PagerAdapter dentro del ViewPager

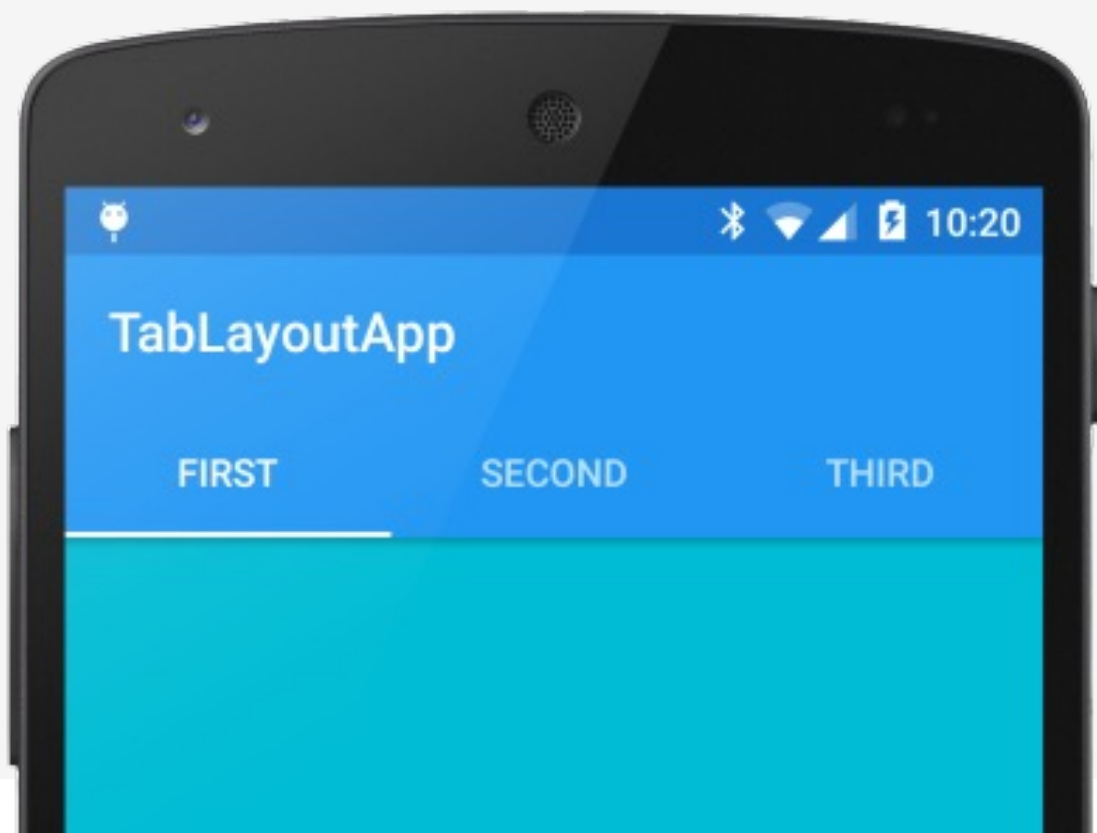
```
11 public class MainActivity extends AppCompatActivity {  
12     ViewPager2 viewPager;  
13     private FragmentStateAdapter pagerAdapter;  
14  
15     @Override  
16     protected void onCreate(Bundle savedInstanceState) {  
17         super.onCreate(savedInstanceState);  
18         setContentView(R.layout.activity_main);  
19         viewPager = findViewById(R.id.viewPager);  
20         pagerAdapter = new MyViewPagerAdapter(this);  
21         viewPager.setAdapter(pagerAdapter);  
22  
23     }  
24 }
```

Pasos para crear una Navegación con desplazamientos



*ViewPager en
Funcionamiento*

Navegación con Pestañas (TabLayout)

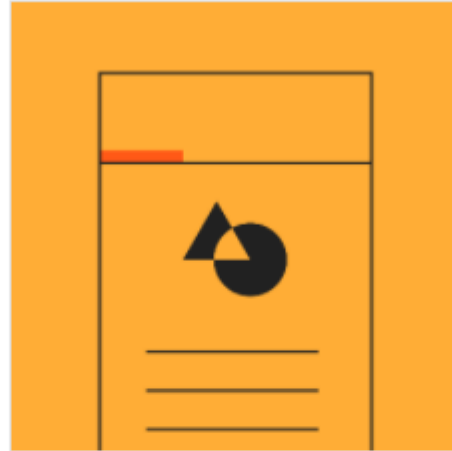


Navegación con Pestañas (Recomendaciones)



Escalable

Como las pestañas pueden desplazarse horizontalmente, una interfaz de usuario puede tener tantas pestañas como sea necesario.



Informativo

Las pestañas organizan el contenido en categorías para ayudar a los usuarios a encontrar fácilmente diferentes tipos de información.

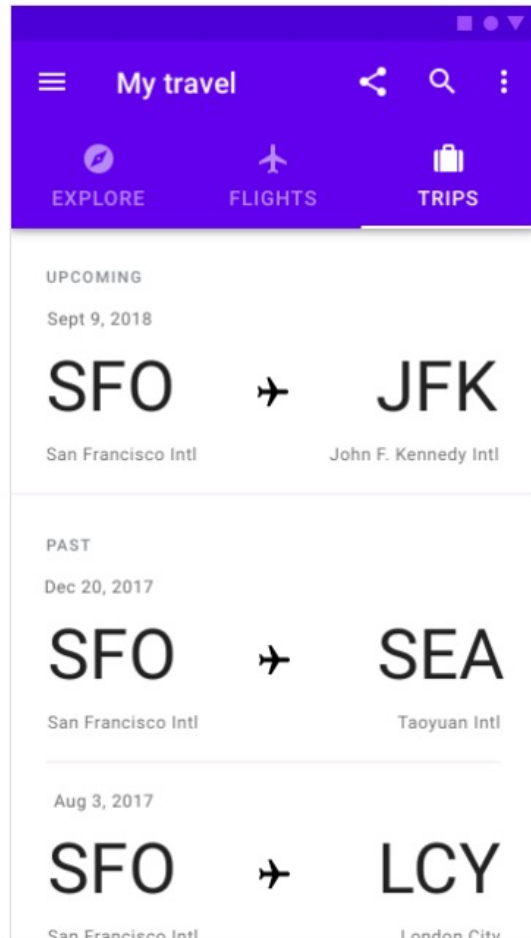


Compañeros

Las pestañas se muestran una al lado de la otra como pares, en categorías de igual importancia.

Las pestañas se pueden combinar con otros componentes como las **barras superiores** de la aplicación o anidar en componentes como **tarjetas y hojas**.

Navegación con Pestañas (Recomendaciones)



hacer

Las etiquetas de tabulación pueden incluir iconos y texto. Las etiquetas de texto deben ser cortas.



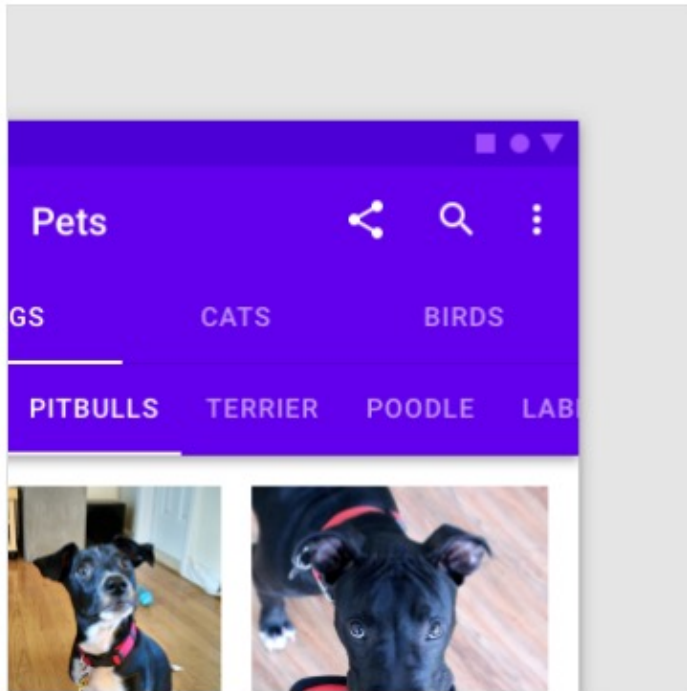
No

No utilice pestañas para moverse a través del contenido secuencial que necesita ser leído en un orden determinado.

- Cada pestaña debe tener **distinto contenido** al de los demás.
- Por ejemplo, las pestañas pueden presentar diferentes secciones de noticias, diferentes géneros de música o diferentes temas de documentos.

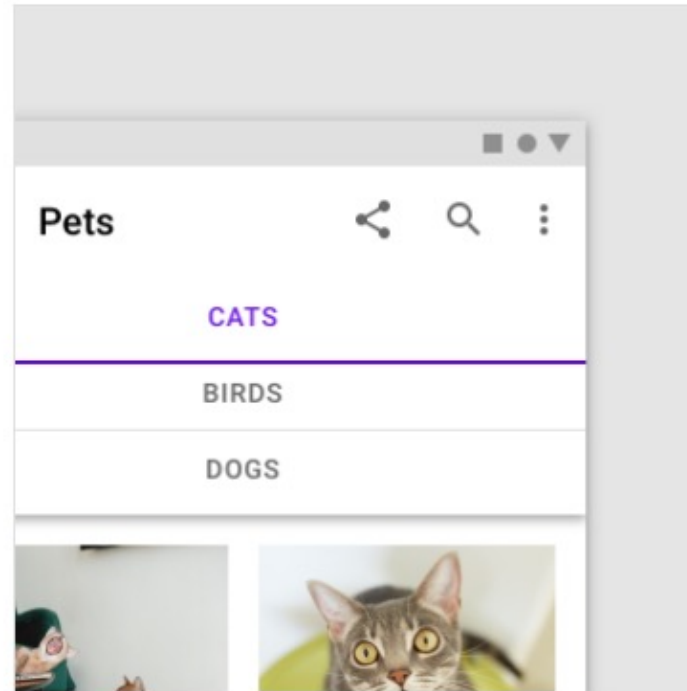
Navegación con Pestañas (Recomendaciones)

- Colocar las pestañas por encima del contenido



Precaución

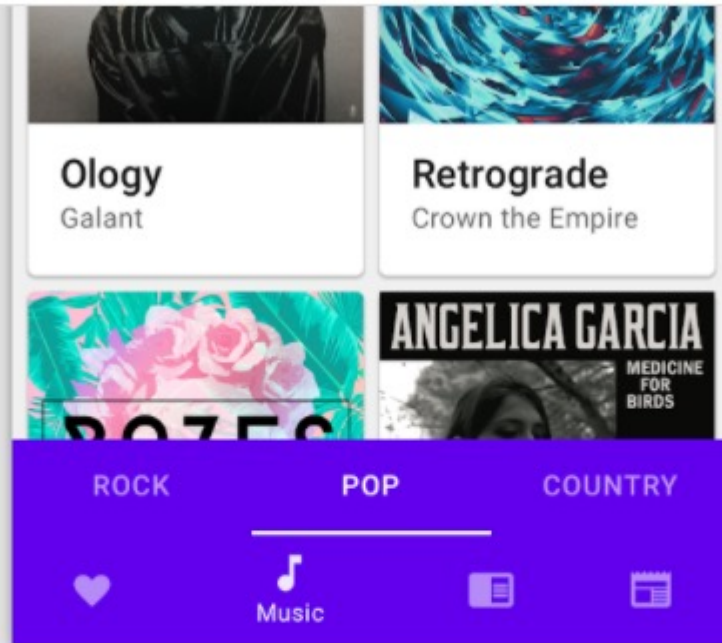
Evite anidar una pestaña dentro de otra pestaña.



No

No coloques pestañas una debajo de la otra.

Navegación con Pestañas (Recomendaciones)



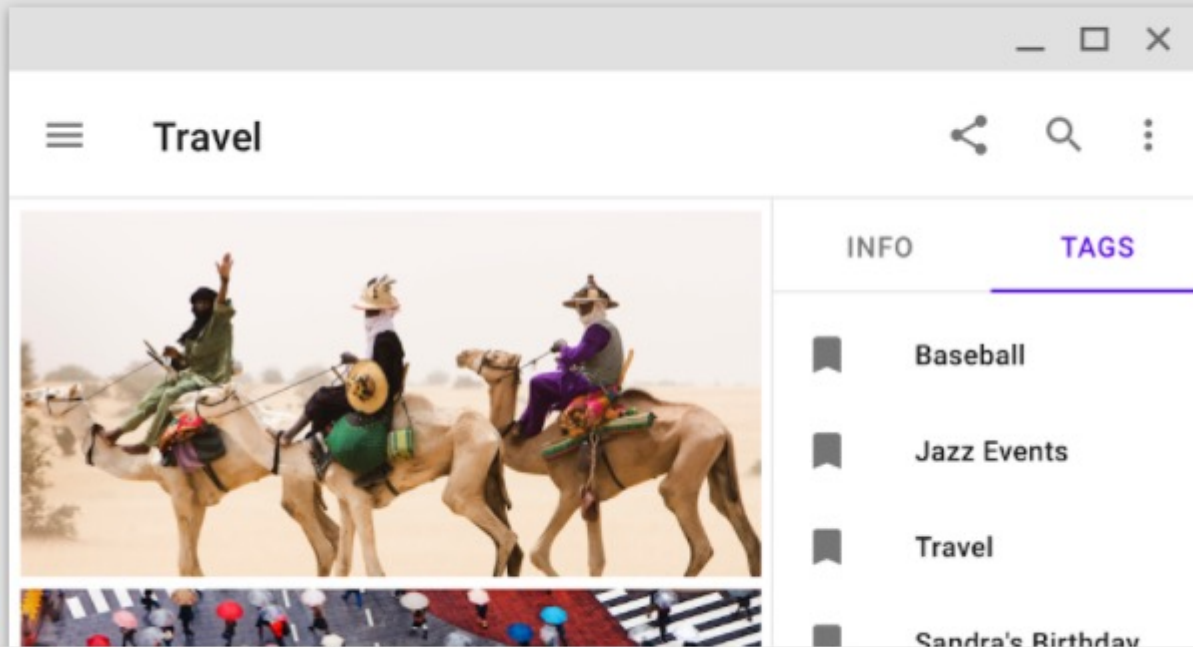
Colocar las pestañas por encima del contenido

No

No adjunte pestañas a la navegación inferior, ya que puede causar confusión sobre qué acción o pestaña controla qué contenido.

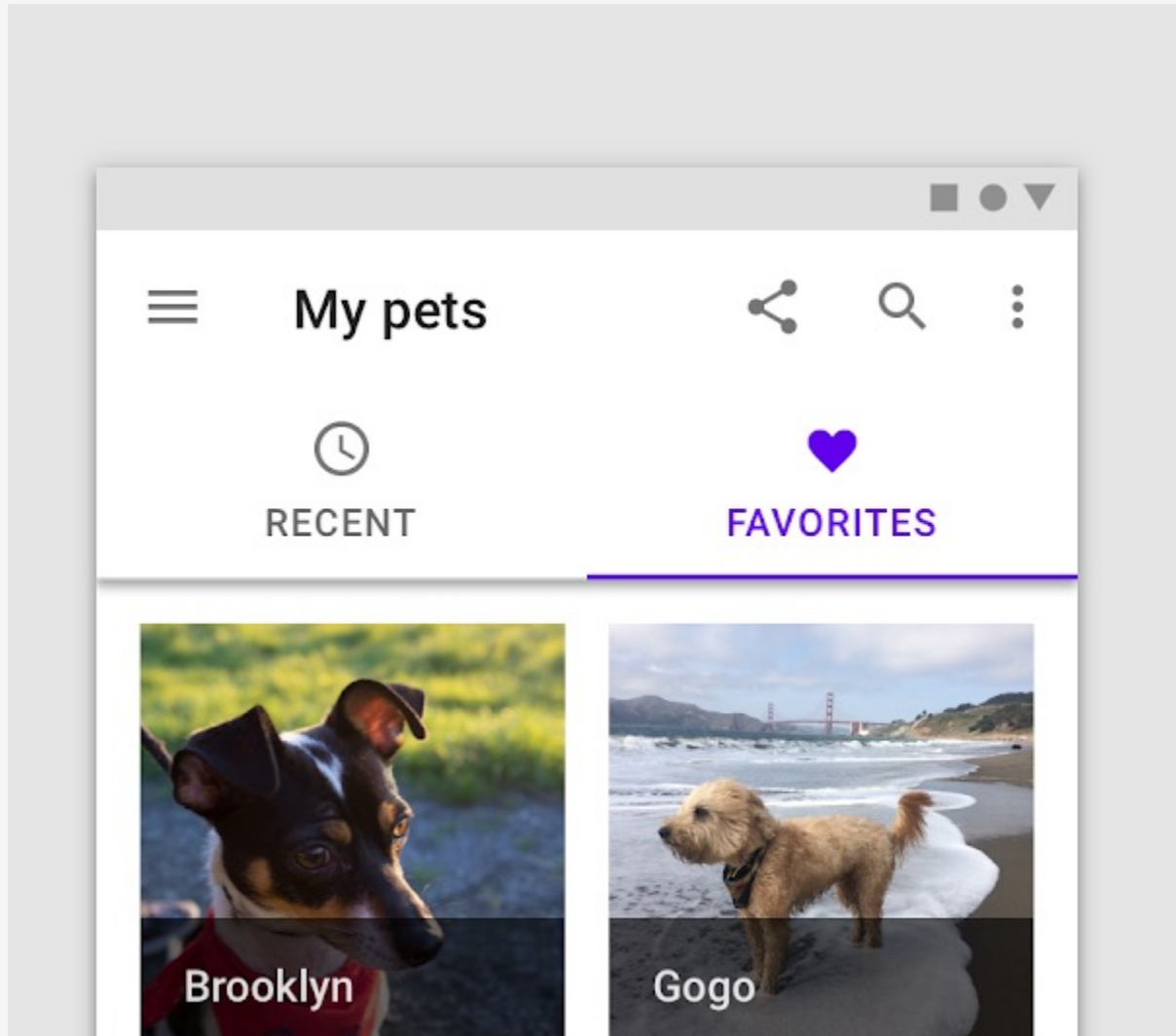
Navegación con Pestañas (Recomendaciones)

- Las pestañas se pueden incrustar en una región de interfaz de usuario



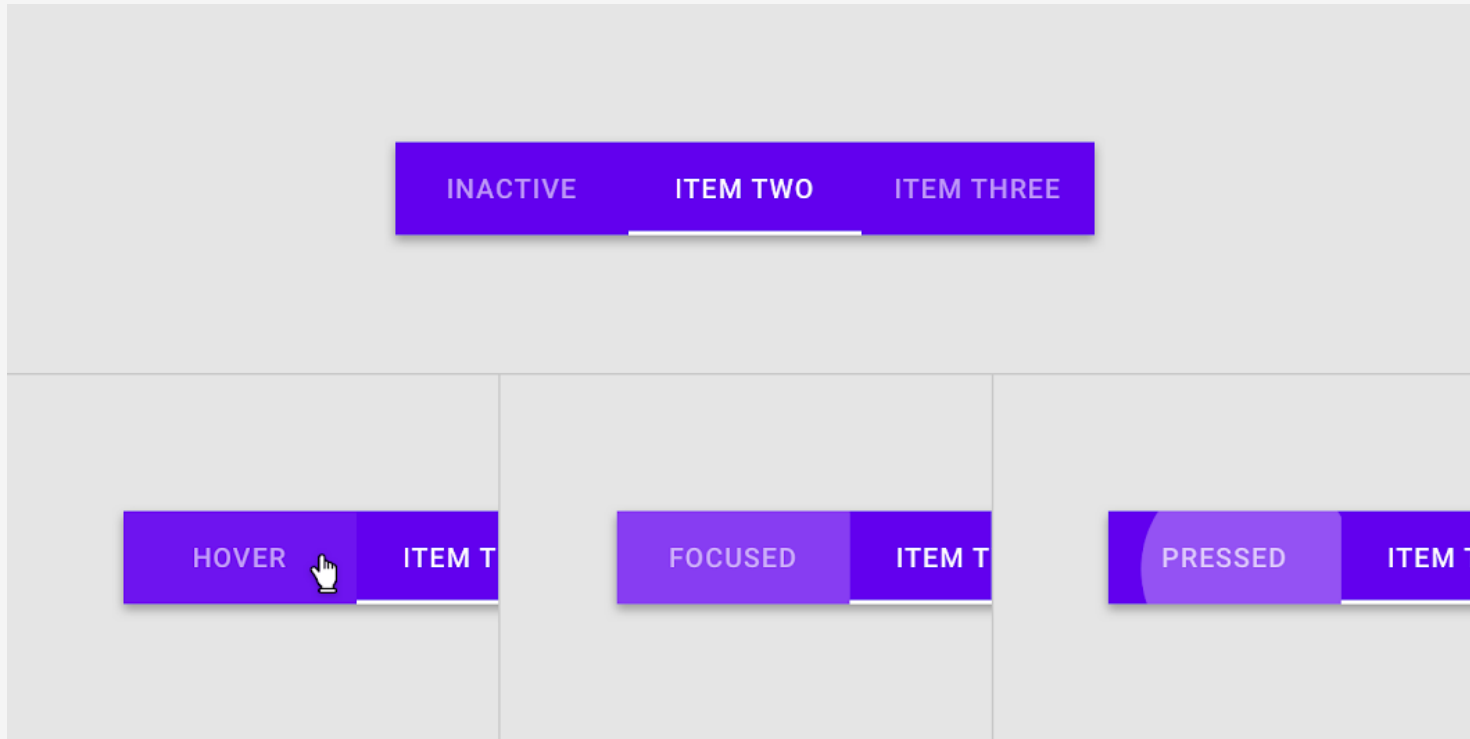
Pestañas incrustadas en una columna, cuyo contenido pueden controlar cuando se toca. Reducido al 62,5%

Navegación con Pestañas (Recomendaciones)



- Una barra de pestañas fija con iconografía y etiquetas

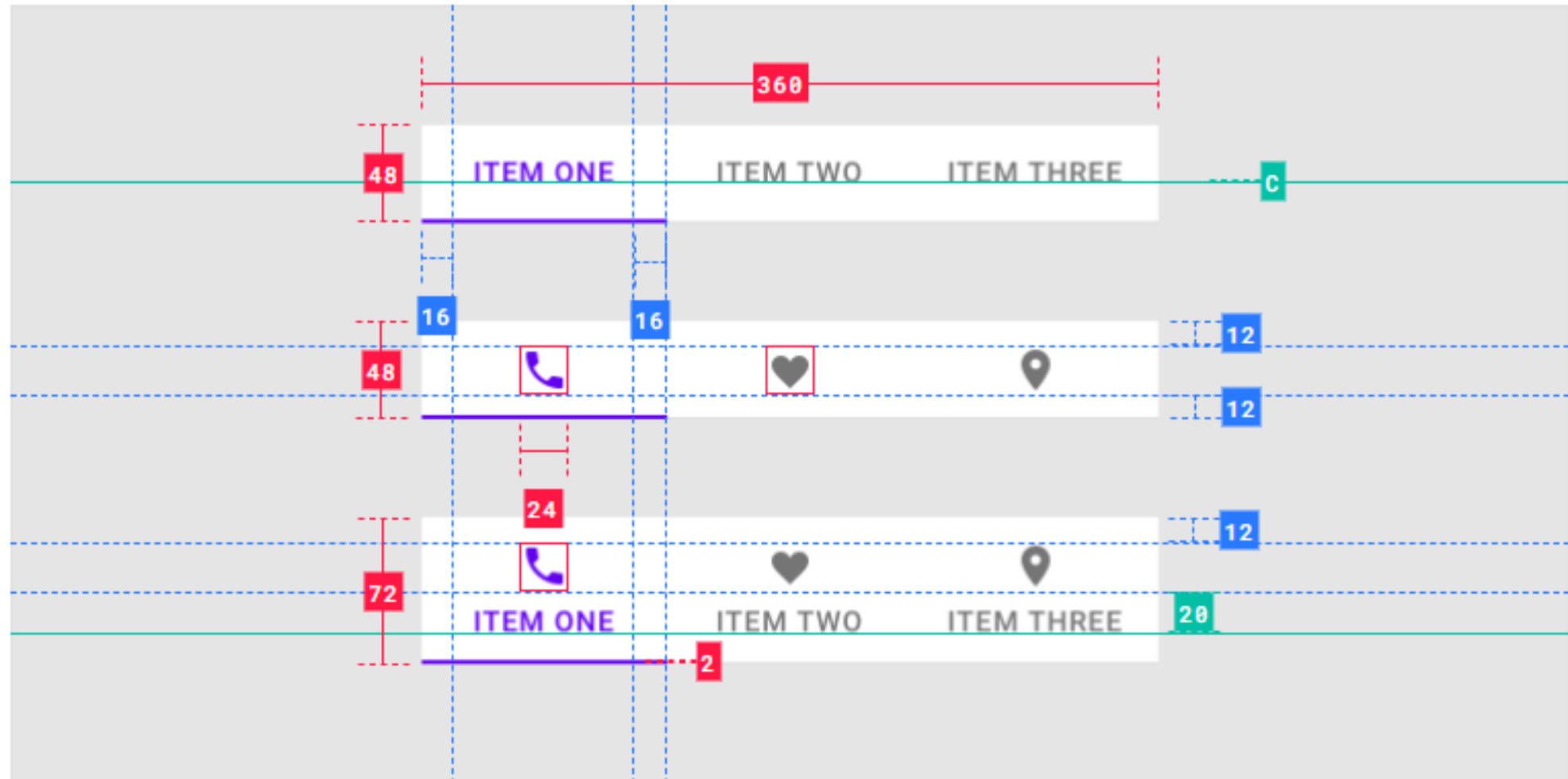
Navegación con Pestañas (Recomendaciones)



- Los estados de desplazamiento, enfoque y prensado de una pestaña habilitada

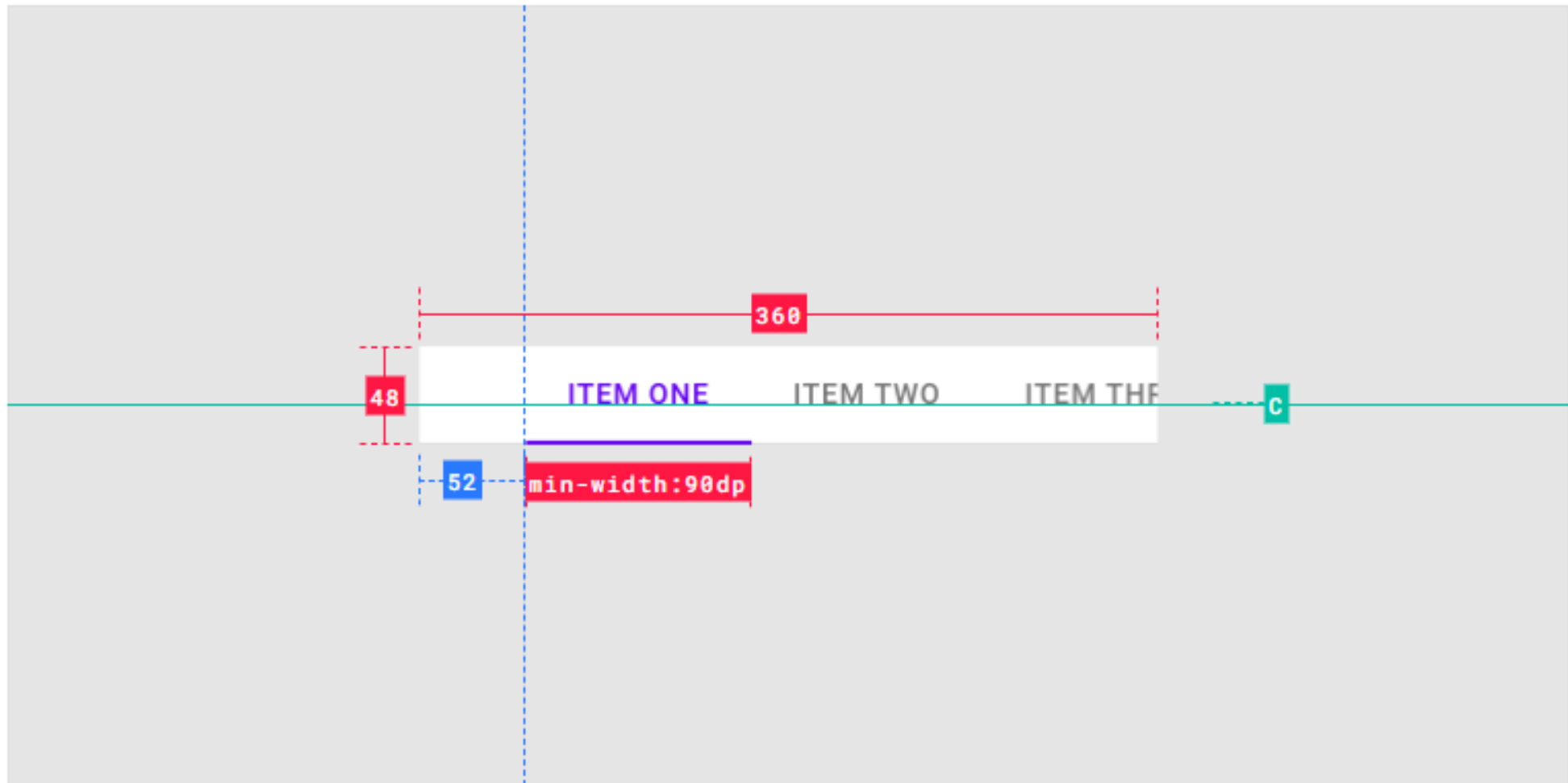
Navegación con Pestañas (Recomendaciones)

Pestañas fijas



Navegación con Pestañas (Recomendaciones)

Pestañas desplazables



Pasos para implementar una navegación de pestañas

1. Agregar el `TabLayout` y `viewPager` al Layout Principal
2. Crear los fragmentos que mostrará el `TabLayout`
3. Implementar un *`PagerAdapter`* para el `viewPager`
4. Integrar el *`ViewPager`* con el *`TabLayout`*

Pasos para implementar una navegación de pestañas

```
<com.google.android.material.tabs.TabLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content">  
  
    ...  
  
</com.google.android.material.tabs.TabLayout>
```

Pasos para implementar una navegación de pestañas

```
<com.google.android.material.tabs.TabLayout
...>

<com.google.android.material.tabs.TabItem
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/text_label_1"
/>

<com.google.android.material.tabs.TabItem
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/text_label_2"
/>

...

</com.google.android.material.tabs.TabLayout>
```

Uso de pestañas con ViewPager

```
<androidx.viewpager2.widget.ViewPager2
    android:id="@+id/viewPager"
    android:layout_width="0dp"
    android:layout_height="0dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tabBar" />
```

Uso de pestañas con ViewPager

```
activity_main.xml ×
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      android:orientation="vertical">
9
10     <com.google.android.material.tabs.TabLayout...>
11
36
37     <androidx.viewpager2.widget.ViewPager2...>
45
46 </androidx.constraintlayout.widget.ConstraintLayout>
```

Pasos para implementar una navegación de pestañas

```
tabBar.addTabSelectedListener(new TabLayout.OnTabSelectedListener() {  
    @Override  
    public void onTabSelected(TabLayout.Tab tab) {  
        viewPager.setCurrentItem(tab.getPosition());  
    }  
  
    @Override  
    public void onTabUnselected(TabLayout.Tab tab) {  
    }  
  
    @Override  
    public void onTabReselected(TabLayout.Tab tab) {  
    }  
});
```

Pasos para implementar una navegación de pestañas

```
ViewPager2 viewPager = findViewById(R.id.viewPager);
viewPager.registerOnPageChangeCallback(new ViewPager2.OnPageChangeCallback() {
    @Override
    public void onPageSelected(int position) {
        //super.onPageSelected(position);
        tabBar.selectTab(tabBar.getTabAt(position));
    }
});
```

NavigationComponent



Ventajas de usar Navigación Component

1. Manejo automático de transacciones de fragmentos
2. Control de forma predeterminada la acción de volver atrás
3. Comportamientos predeterminados para animaciones y transiciones
4. Es posible visualizar y editar el flujo de navegación de una aplicación

Partes de un componente de navegación



Partes de un componente de navegación

Gráfico de navegación (nuevo recurso XML):

- 1. Es un recurso que contiene toda la información relacionada con la navegación en una ubicación centralizada.*
- 2. incluye todos los lugares de la aplicación, conocidos como destinos, y posibles rutas de acceso que un usuario podría tomar a través de la aplicación.*

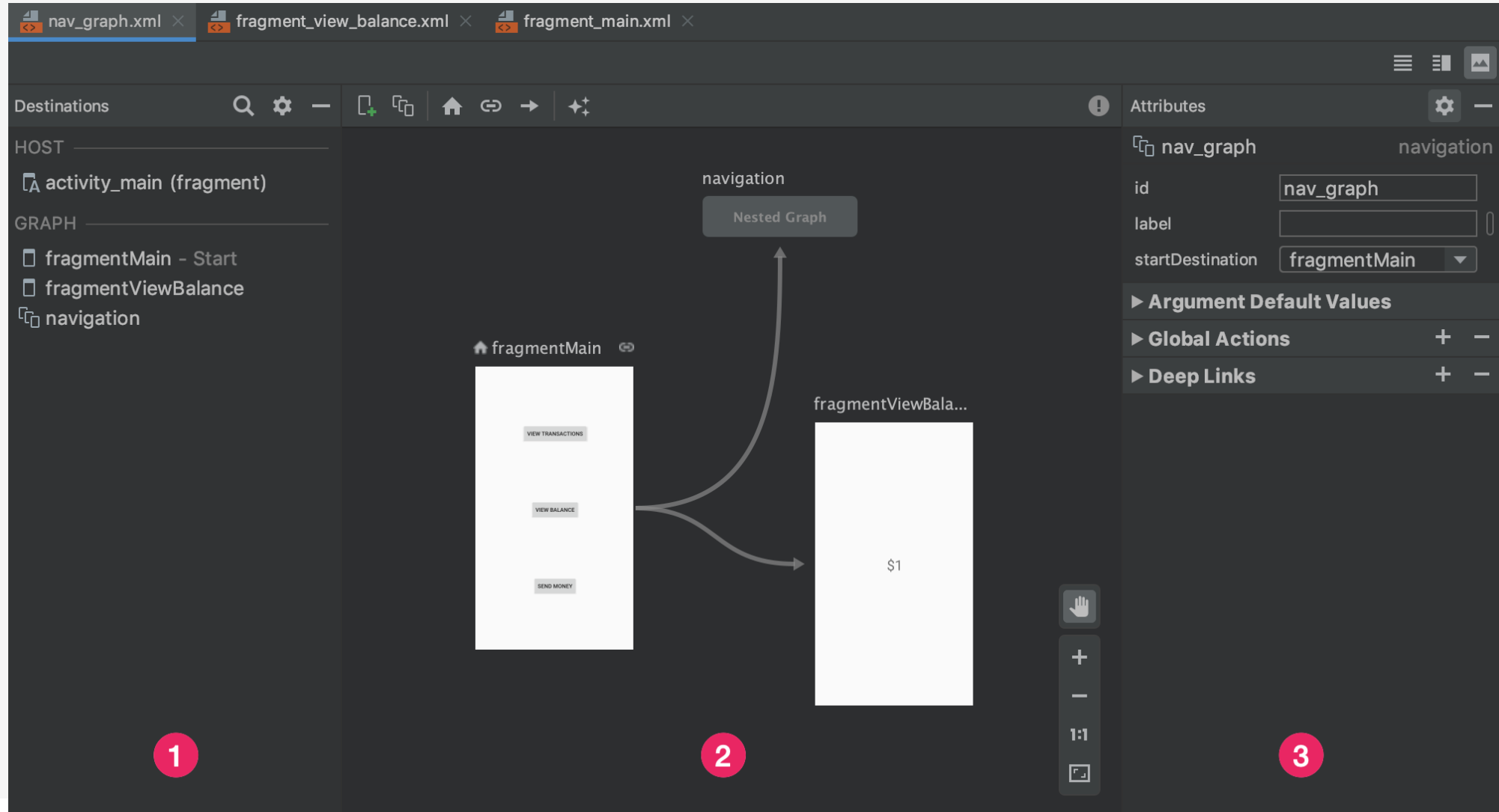
Partes de un componente de navegación

Gráfico de navegación (nuevo recurso XML):

```
<?xml version="1.0" encoding="utf-8"?>
<navigation xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  xmlns:android="http://schemas.android.com/apk/res/android"
  app:startDestination="@id/blankFragment">
  <fragment
    android:id="@+id/blankFragment"
    android:name="com.example.cashdog.cashdog.BlankFragment"
    android:label="fragment_blank"
    tools:layout="@layout/fragment_blank" >
    <action
      android:id="@+id/action_blankFragment_to_blankFragment2"
      app:destination="@id/blankFragment2" />
  </fragment>
  <fragment
    android:id="@+id/blankFragment2"
    android:name="com.example.cashdog.cashdog.BlankFragment2"
    android:label="fragment_blank_fragment2"
    tools:layout="@layout/fragment_blank_fragment2" />
</navigation>
```

Partes de un componente de navegación

Gráfico de navegación (nuevo recurso XML):



Partes de un componente de navegación

NavHostFragment (vista XML de diseño):

1. *Es un widget especial que se agrega al diseño (activity_main.xml).*
2. *Este componente se encarga de mostrar los diferentes destinos del gráfico de navegación.*

Partes de un componente de navegación

NavHostFragment (vista XML de diseño):

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <androidx.appcompat.widget.Toolbar
        .../>

    <androidx.fragment.app.FragmentContainerView
        android:id="@+id/nav_host_fragment"
        android:name="androidx.navigation.fragment.NavHostFragment"
        android:layout_width="0dp"
        android:layout_height="0dp"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintBottom_toBottomOf="parent"

        app:defaultNavHost="true"
        app:navGraph="@navigation/nav_graph" />

    <com.google.android.material.bottomnavigation.BottomNavigationView
        .../>

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

Partes de un componente de navegación

NavController (objeto Kotlin/Java):

1. *Es un **objeto** que realiza un **seguimiento de la posición** actual dentro del gráfico de navegación.*
2. *Orquesta el intercambio de contenido de destino en NavHostFragment a medida que se mueve a través de un gráfico de navegación.*

Partes de un componente de navegación

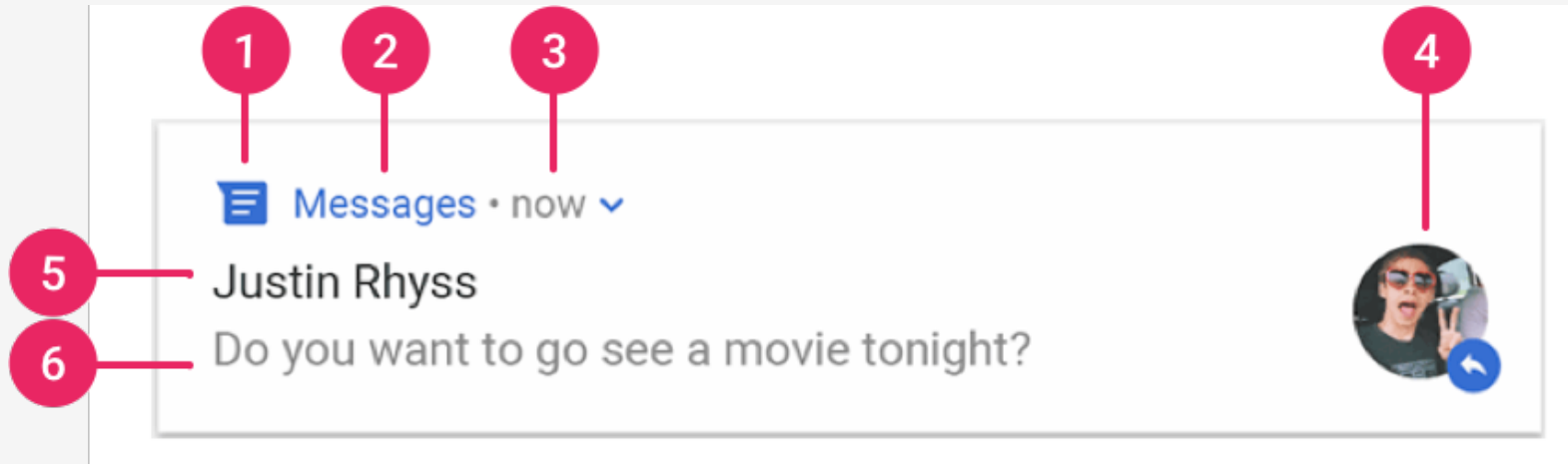
NavController (objeto Kotlin/Java):

Java:

- `NavHostFragment.findNavController(Fragment)`
- `Navigation.findNavController(Activity, @IdRes int viewId)`
- `Navigation.findNavController(View)`

```
public class MainActivity extends AppCompatActivity {  
    NavController navController;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        NavHostFragment navHostFragment = (NavHostFragment) getSupportFragmentManager().findFragmentById(R.id.my_nav_host_fr);  
        NavController navController = navHostFragment.getNavController();  
    }  
}
```

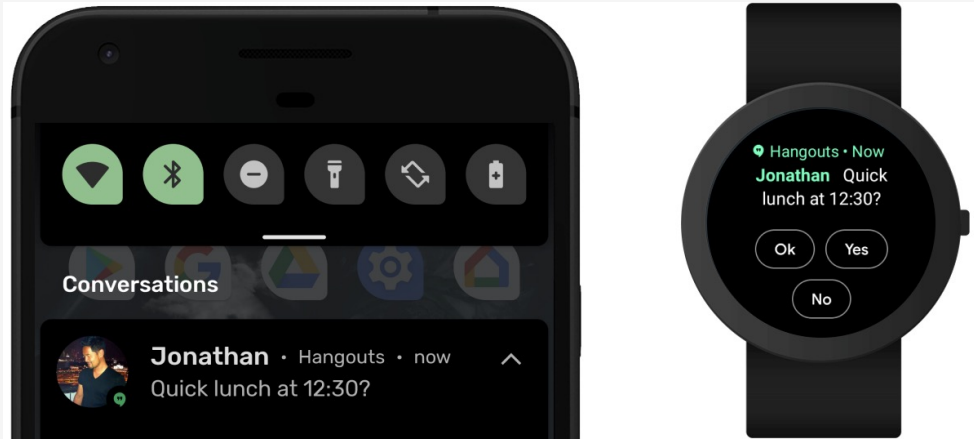
Notificaciones



- 1 Ícono pequeño: Es obligatorio y se establece con `setSmallIcon()`.
- 2 Nombre de la app: Proporcionado por el sistema.
- 3 Marca de tiempo: El sistema la proporciona, pero puedes anularla con `setWhen()` o bien ocultarla con `setShowWhen(false)`.
- 4 Ícono grande: Es opcional (en general, se usa solo para fotos de contacto; no lo uses para el ícono de la app) y se configura con `setLargeIcon()`.
- 5 Título: Es opcional y se establece con `setContentTitle()`.
- 6 Texto: Es opcional y se establece con `setContentText()`.

- Es un mensaje que muestra Android fuera de la IU de la app.
- Proporciona al usuario recordatorios, mensajes y otra información puntual de la app.

Tipos de notificaciones



 Gmail • 8 min ▾

Justin Rhyss
Movie night



Expanded

 Gmail • 8 min ▲

Justin Rhyss
Movie night



Hey, do you have any plans for tonight? I was thinking a few of us could go watch a movie at the theater nearby since there won't be much going on for the next couple of weeks. There are some great options at 6 and 7pm, but whatever works best for you. If you have any suggestions for dinner beforehand hit reply!

REPLY ARCHIVE

 Messages • now ▲

Justin Rhyss
Do you want to go see a movie tonight?



REPLY ARCHIVE

Pasos para crear una Notificación

- 1. Crear un Canal de Notificación**
- 2. Definir un ícono**
- 3. Definir un título**
- 4. Definir el contenido de la notificación.**
- 5. Mostrar la notificación**

Gracias por su atención