# Pestañas

Asociamos cada pestaña (TabItem) con un fragment

Es necesario un viewPager, que sería parecido a un Recycler View.

Cada tab es un layout con un texto en medio como este:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente con confianza media

## Activity\_main

Una vez creados los fragments, gestionaremos el cambio de un tab a otro desde el activity\_main y el viewPager. El Layout que usamos para los fragments de las pestañas se llama TabLayout. Este va dentro de cada fragment.

Texto

Descripción generada automáticamente

En primer lugar, tiene que haber un **Adapter**, como con el RecyclerView, para poder trabajar con 3 pestañas.

Necesitamos redefinir dos métodos, uno que nos devuelve cuantas pestañas tenemos, y otro que nos devuelve el fragment correspondiente a cada una de las pestañas.

Texto

Descripción generada automáticamente

También tiene que haber un LayoutManager. En este caso, se llama **TabLayoutMediator**.

Asignamos un nombre a cada una de las pestañas, y luego hace un attach, para relacionarlos/conectarlos. Conecta el TabLayout y el ViewPager.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

# Menus

Primero crear un nuevo directorio de recursos:

Interfaz de usuario gráfica

Descripción generada automáticamente

Los menús no tienen por que pertenecer a un activity, podrían ser también de un fragment.

La diferencia es que no tendremos un fichero Kotlin especifico para este menú, sino que a diferencia de los Fragments y Activities, lo gestionaremos desde el actual.

Hay que crear **New Resource Menu**

## ActionBar

Se recomienda anularla y utilizar la Toolbar.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Y después seguir los pasos que hemos visto anteriormente en el apartado menús en general.

Texto

Descripción generada automáticamente

El menú, debe tener SI O SI un icono vectorial.

Una captura de pantalla de una computadora

Descripción generada automáticamente

Después, en el activity main, hay que poner esto:

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Interesante atributo: actionBarSize:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Esto es para que tenga el mismo tamaño de la actionBar.

Después necesito reimplementar dos métodos:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Aquí se recibe una variable de tipo Item, y con un when, le vamos indicnado los distintos id’s que hemos puesto en el fichero xml del menú.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Falta poner return true.

# Proyecto:

* Activity\_Main con bottom bar menu
  + Fragment por cada opcion del bottom.
    - Fragment con Tab viewPager
      * Fragment por cada pestañas.