

DIFERENTES VENTANAS SEGÚN EL DISPOSITIVO USADO

Vamos a ver como cargar fragmentos según el dispositivo sea una Tablet o un teléfono. Vamos a basarnos en el ejemplo de la documentación de Android sobre fragmentos

<https://developer.android.com/guide/components/fragments?hl=es-419>

En esta vemos que si la pantalla es muy grande podremos cargar dos fragmentos en la misma, pero si es un dispositivo de mano será más efectivo cargar un fragmento y al hacer alguna acción cargar el siguiente fragmento.

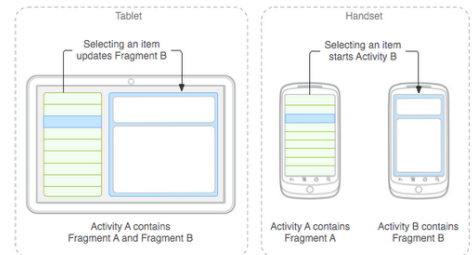


Figura 1: Ejemplo de la manera en que dos módulos de la IU definidos por fragmentos se pueden combinar en una actividad para un diseño de tablet y se presentan por separado para un diseño de teléfono.

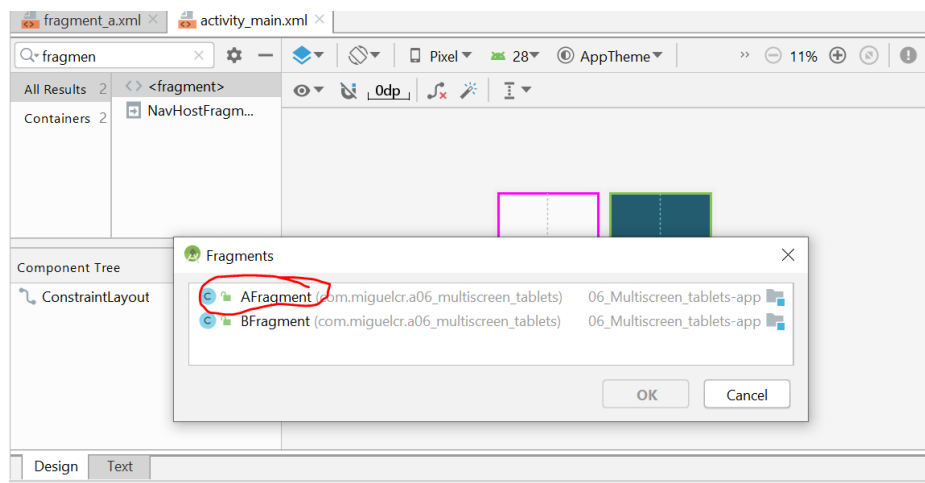
Paso 1:

Creamos un emulador de una Tablet (o usamos una tablet).

Vamos a crear un proyecto como hasta ahora orientado a un terminal móvil con una Empty Activity.

Creamos dos Fragment Blank (AFragment y BFragment) en ambos desmarcar las opciones que vienen marcadas ya que nos vamos a usar en este caso el código que generan.

En el activity_main.xml cambiamos el TextView por un <fragment> con ancho y alto match_parent, es decir fragmentos estáticos que para este ejemplo van perfectamente. Al arrastrar un elemento de tipo fragment nos dice que seleccionemos que fragmento cargar en él, y nosotros le vamos a decir que por defecto sea el AFragment.java



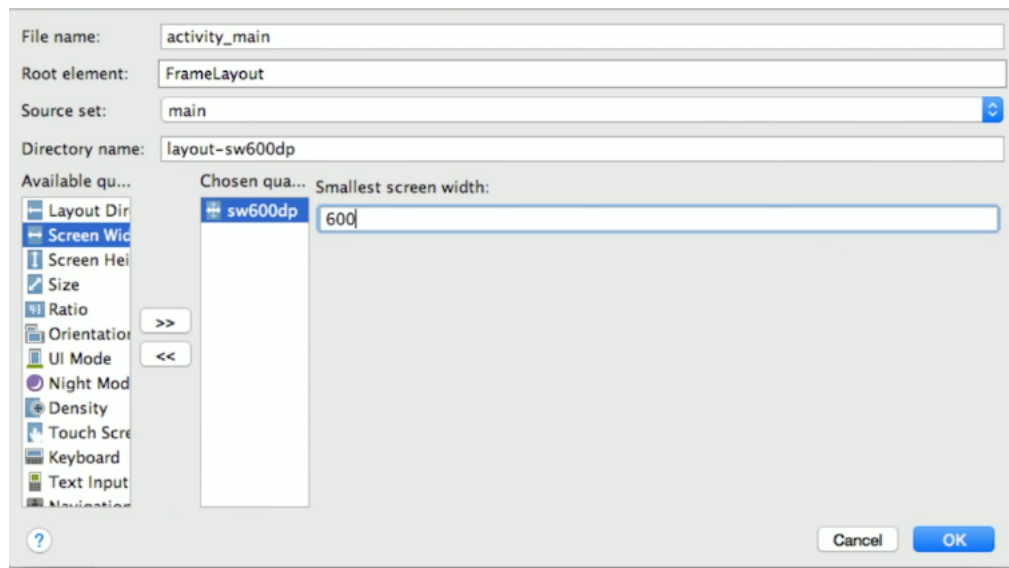
En el layout de cada uno de los fragmentos creados les cambiamos el fondo y si queremos el texto de cada uno de ellos.

Si ejecutamos en cada uno de los emuladores vemos que en el móvil se ve el fragmento A perfecto pero en la Tablet es mucha pantalla para un solo fragmento, sería mejor ver los dos fragmentos de golpe.

Paso 2:

Igual que en el ejemplo anterior voy a crear un nuevo activity_main para que cargue uno u otro en función de la anchura de la pantalla.

Sobre la carpeta layout hacer click con segundo botón del ratón y seleccionar new→Layout Resource file y dentro de esta seguir las indicaciones que se muestran en la siguiente imagen:



En este nuevo activity_main cambiamos el texto, por ejemplo “Ancho de al menos 600dp”

Si probamos esto vemos que según el dispositivo en el que se ejecute va a cargar el fragmento en el móvil y mostrará en la Tablet este último activity_main con la frase que he puesto.

Como vemos entonces, en función de la densidad de la pantalla cargará un activity_main, por tanto, en este último vamos a cargarle dos fragment, uno con cada uno de los fragmentos que tenemos, para ello y para que se muestren de forma ordenada, hagamos que el contenedor sea por ejemplo un LinearLayout y arrastremos los dos fragmentos como en el caso anterior.

A continuación, en el MainActivity.java vamos a hacer que cargue un main_activity u otro, para ello, en el onCreate definimos un layoutContenedor de tipo LinearLayout que es el contenedor que tenemos en nuestro segundo main_activity.xml, si la densidad de la pantalla es de 600dp o más se cargará, teniendo un valor distinto de nulo, caso contrario si la densidad es menor. Teniendo esto podemos ejecutar un condicional, si es null sabemos que estamos en una Tablet (y nos mostrará los dos fragmentos que tiene cargados el correspondiente activity_main.xml) y si no, estaremos en un dispositivo móvil y nos cargará solo el primer fragmento con el activity_main.xml correspondiente.

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    LinearLayout layoutContenedor = (LinearLayout)findViewById(R.id.contenedor);  
  
    if(layoutContenedor!=null) {  
        // Tablet  
        Toast.makeText(context: this, text: "Te encuentras en una tablet", Toast.LENGTH_SHORT).show();  
    } else {  
        // Móvil  
        Toast.makeText(context: this, text: "Te encuentras en un móvil de mano", Toast.LENGTH_SHORT).show();  
    }  
}  
}
```