

Evidencia Lunes 26 de Julio (día 01 / Semana 14)

Leonardo Rodenas Escobar

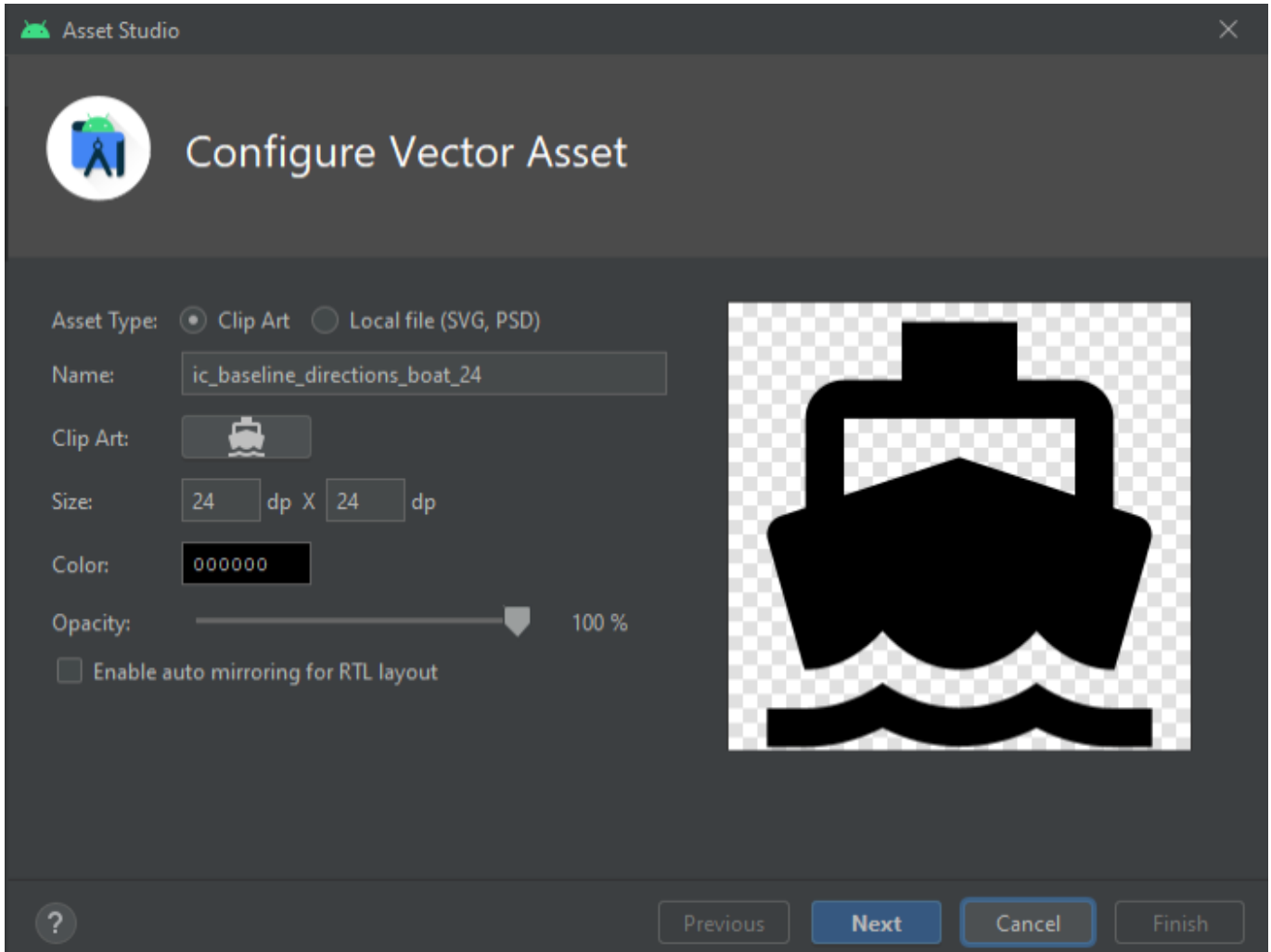
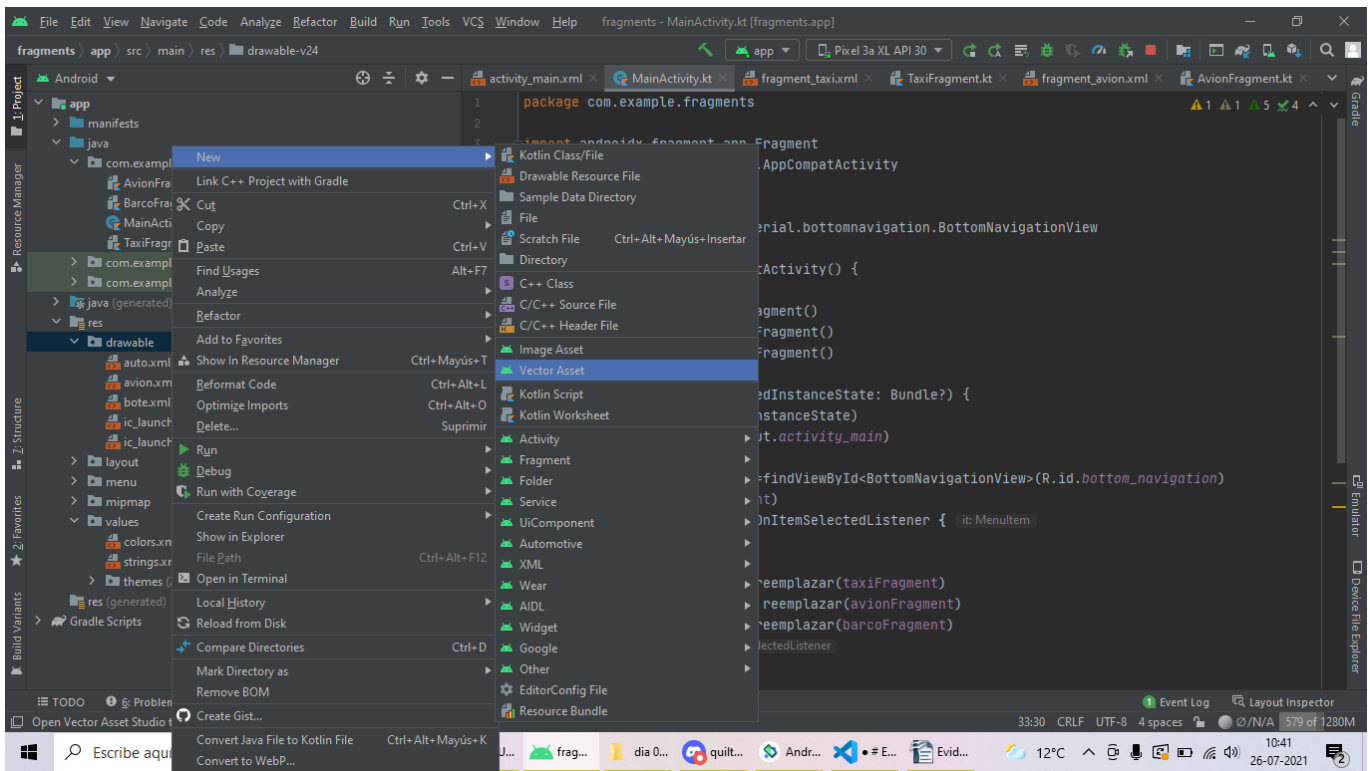
Reflexión:

Esta semana se comienza con algo mas interesante que la anterior, razón de que sea un poco más motivante el aprendizaje. De esta manera los fragmentes vistos el dia de hoy, junto con la informacion buscada por mi dias anteriores, me ayuda a plantearme de mejor manera como realizar las apps y optimizar los recursos del teléfono a medida que desarrollo. El ejemplo visto en clases es básico, pero bueno para empezar, además de que me resultó sin mayores complicaciones.

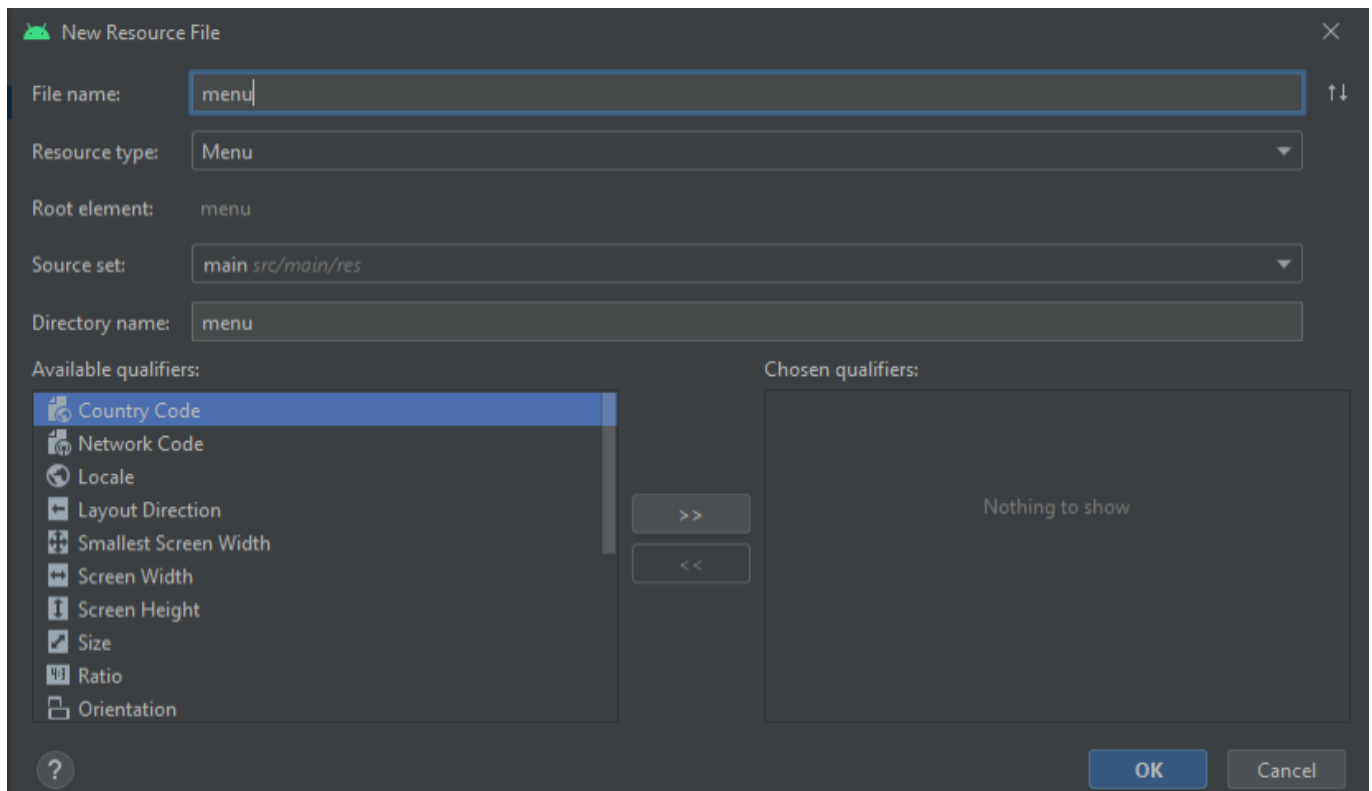
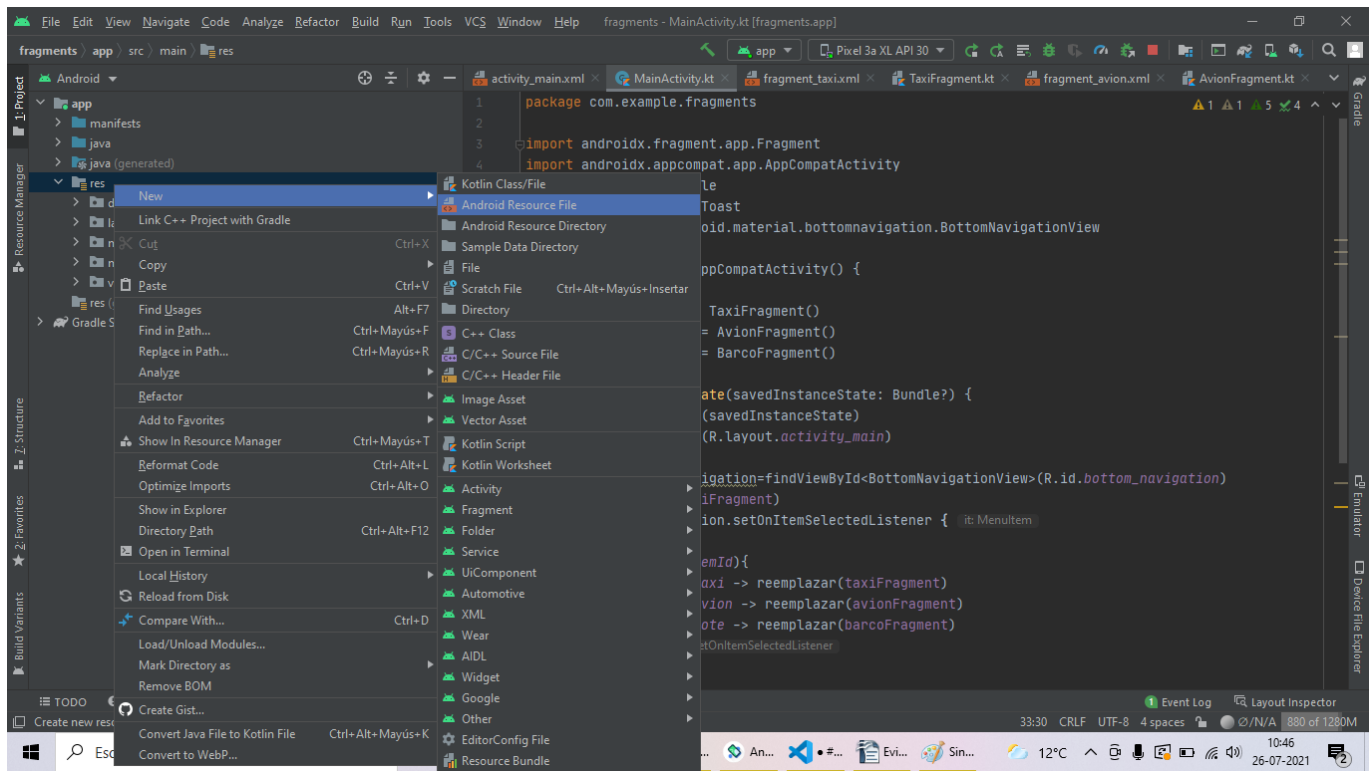
Actividad 01:

Realizar Fragments y Action Bar según ejemplo visto en clases.

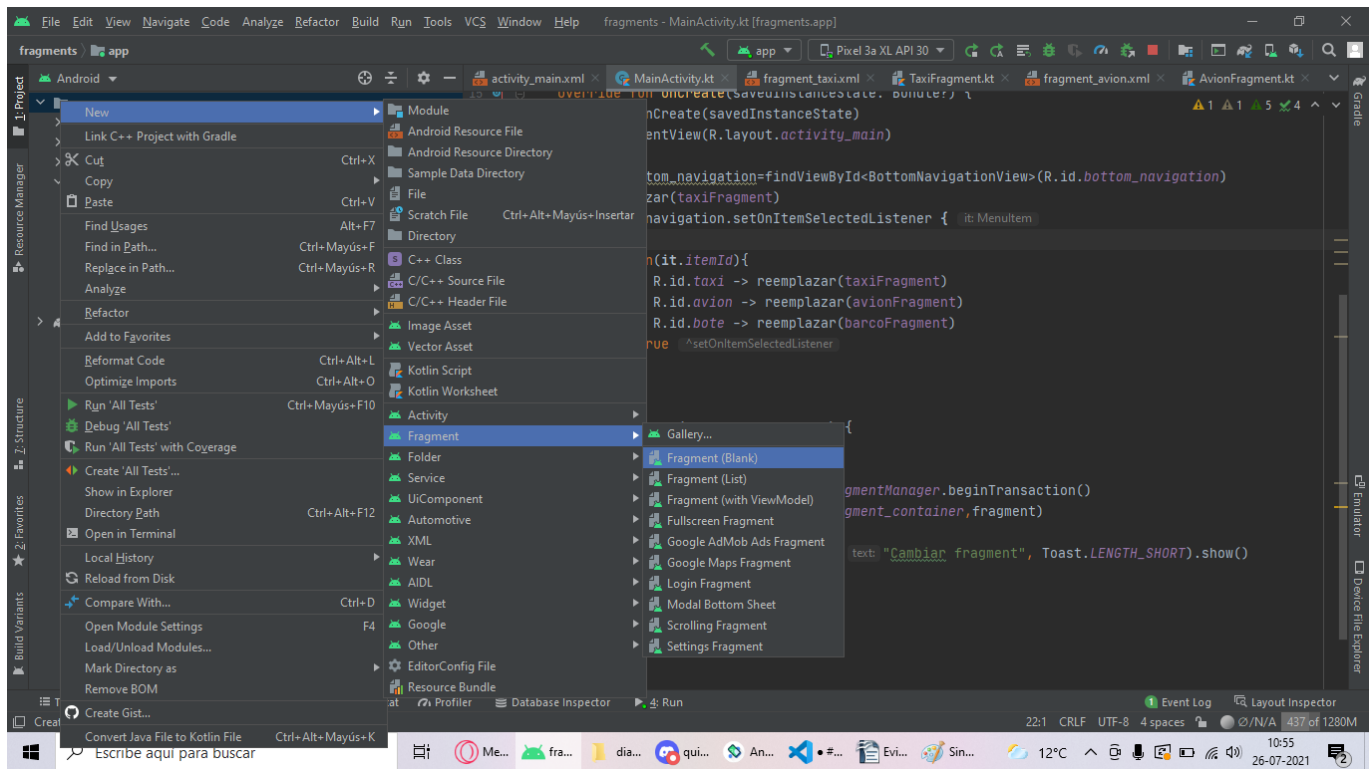
- Importamos las imágenes a usar en los botones de la siguientes manera: Project/res/drawable → New Vector asset (acá se selecciona la imagen, se le da color y nombre, además de otros atributos si se necesita y se guarda en la carpeta especificada, en el ejemplo se hace con 3 imágenes, un auto, un avión y un barco)



- Creamos un menú → Desde la carpeta res, new android resource file, se crea un archivo tipo menú



- Luego, se crean los 3 fragments dentro de la carpeta app para cada uno de los componentes que van a estar presentes en la Action Bar (la barrita de abajo de las app), notar que el TextView de cada fragment se cambia a uno que indique a que se le dio click (Un auto, un avión o un barco).



- Con los fragments creados, se da la interfaz al menu en el xml de diseño.

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <item
        android:id="@+id/taxi"
        android:title="Item"
        android:icon="@drawable/auto"/>

    <item
        android:id="@+id/avion"
        android:title="Item"
        android:icon="@drawable/avion" />

    <item
        android:id="@+id/bote"
        android:title="Item"
        android:icon="@drawable/bote"/>

</menu>
```

- Luego, en el main Activity (.xml) se crea la interfaz que contendrá la Action Bar de la siguiente manera:

```
<?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">

<com.google.android.material.bottomnavigation.BottomNavigationView
    android:id="@+id/bottom_navigation"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintLeft_toRightOf="parent"
    app:menu="@menu/menu"
    android:background="@color/teal_700"
    app:itemIconTint="@color/white"
    app:itemTextColor="@color/white"/>

    <FrameLayout
        android:id="@+id/fragment_container"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        app:layout_constraintBottom_toTopOf="@id/bottom_navigation"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

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

Obteniendo el siguiente resultado gráfico:



- Finalmente, desde la activity main (.kt) se añade la lógica, escribiendo el código que permitirá funcionar estos fragments de la siguiente manera:

```
package com.example.fragments

import androidx.fragment.app.Fragment
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.google.android.material.bottomnavigation.BottomNavigationView

class MainActivity : AppCompatActivity() {

    val taxiFragment = TaxiFragment()
    val avionFragment = AvionFragment()
    val barcoFragment = BarcoFragment()

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val bottom_navigation=findViewById<BottomNavigationView>
(R.id.bottom_navigation)
        reemplazar(taxiFragment)
        bottom_navigation.setOnItemSelectedListener {
```

```
        when(it.itemId){
            R.id.taxi -> reemplazar(taxiFragment)
            R.id.avion -> reemplazar(avionFragment)
            R.id.bote -> reemplazar(barcoFragment)
        };true
    }
}

fun reemplazar (fragment: Fragment) {

    if (fragment !=null){
        val transaccion = supportFragmentManager.beginTransaction()
        transaccion.replace(R.id.fragment_container,fragment)
        transaccion.commit()
        Toast.makeText(this, "Cambiar fragment", Toast.LENGTH_SHORT).show()
    }

}

}
```

Así, después de estos pasos lo que se obtiene es que al dar click en cada uno de los botones de la Action Bar, dentro de una misma activity, se cambiará por cada fragmente del botón a un mensaje u otro, sin necesidad e hacer un Intent o generar más activitys.



Así concluye mi avance de hoy 😊, muchas gracias!!!