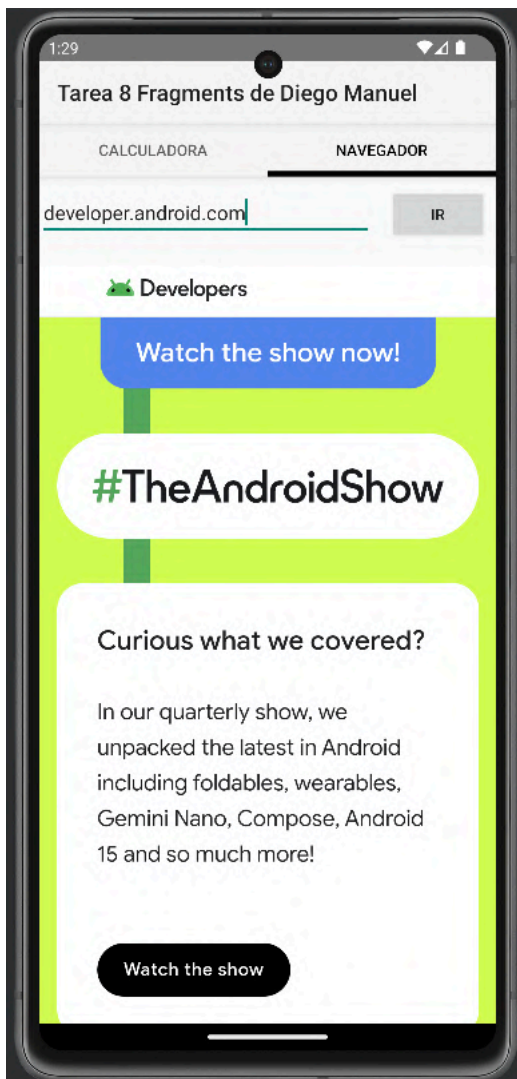


# Tarea 8 para Programación Multimedia y Dispositivos Móviles



Diego Manuel Carrasco Castañares

**Enunciado.**

Tarea UT08: Fragments Calculadora y WebView alternos.

Crear una aplicación que incorpore dos fragments, uno relativo al uso de una calculadora y otro relativo al uso de un WebView.

Inicialmente se cargará este último. El aspecto de la aplicación será el mostrado en la figura 1. Si cargamos una página, vemos que se mantiene la zona superior (figura 2).

Figura 1. Aspecto inicial



Figura 2. Carga de una página



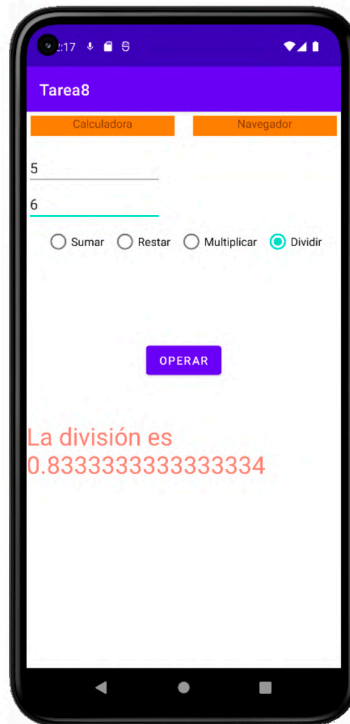
Si ahora clicamos en el botón superior relativo a la calculadora, la apariencia será la mostrada en la figura 3.

Vemos que igualmente se mantiene invariable la parte superior. La calculadora debe ser operativa, por ejemplo, para el caso de división de 5 entre 6 obtenemos el resultado mostrado en la figura 4.

Figura 3. Clic en botón Calculadora



Figura 4. División de 5 entre 6



Ahora podremos volver de nuevo al fragmento de navegación (se cargará de nuevo) tal y como se muestra en la figura 5.

Figura 5. Volver al fragmento



**Criterios de puntuación. Total 10 puntos.**

La valoración total de la tarea es de 10 puntos repartidos del siguiente modo:

- ✓ Aspecto: la aplicación se muestra de un modo similar al mostrado. 1 punto.
- ✓ Funcionamiento de la app:
  - La aplicación muestra un fragmento u otro en función de a elección del usuario: 4 puntos.
  - La aplicación puede abrir una página web: 2 puntos.
  - La calculadora funciona correctamente: 2 puntos.
  - La estructura de archivos del proyecto es acorde a lo pedido: 1 punto.

Puesto que no existe una evaluación por unidades la evaluación se realiza en base a los criterios generales del módulo.

Recursos necesarios para realizar la Tarea.

- ✓ Ordenador con Android Studio instalado y suficientes recursos para ejecutar el emulador.
- ✓ Contenidos de la Unidad, muy importantes los ejercicios resueltos de la misma.
- ✓ Páginas web de los desarrolladores de los sistemas operativos para móviles.

**Consejos y recomendaciones.**

- ✓ Lee atentamente el enunciado y asegúrate de haber entendido lo que has de hacer.
- ✓ Intenta reproducir en tu Android Studio los ejercicios resueltos previamente.
- ✓ No dudes en comentarle a tu tutor o tutora cualquier duda que te pueda surgir.
- ✓ Envíasela a tu tutor o tutora a través del sistema establecido en la plataforma.
- ✓ Las capturas y el contenido de los ficheros deben aparecer en perfecto orden para que esté claro lo que deseas mostrar.

- ✓ Los diseños deben mostrarse lo más parecidos posibles a lo que se pide.
- ✓ Debe llevarse a cabo sobre una versión actual de Android.

**Indicaciones de entrega.**

Una vez realizada la tarea elaborarás un documento PDF.

- ✓ El documento pdf mostrará capturas de la app:
  - Captura del aspecto de tu aplicación al arrancar la primera vez.
  - Captura del aspecto de tu aplicación al cargar el fragment de la web donde se vea una página web que comience con la inicial de tu nombre. Por ejemplo, si te llamas Daniel puedes cargar <https://developer.android.com/>
  - Capturas del aspecto de tu aplicación en modo calculadora realizando operaciones.
- ✓ Añade el contenido de los ficheros editados: java, xml, archivo de manifiesto, etc.
- ✓ Las capturas que no sean de tu aplicación, sino de tu Android Studio, deben mostrar el fondo de pantalla y la fecha y hora de la barra de estado de tu sistema operativo para garantizar la originalidad del material.

También se deberá entregar la carpeta comprimida del proyecto en formato .zip (File → Export → Export to Zip file...)

Entrega los archivos a través de la plataforma. Se nombrarán siguiendo las siguientes pautas:

apellido1\_apellido2\_nombre\_PMDM08\_Tarea

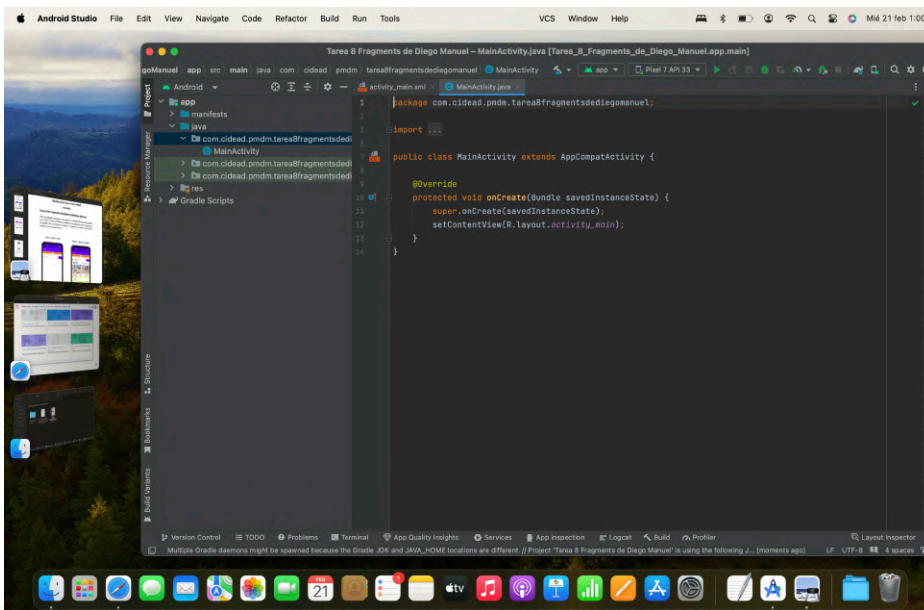
Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños.

Así por ejemplo la alumna Begoña Sánchez Mañas para la séptima unidad del MP

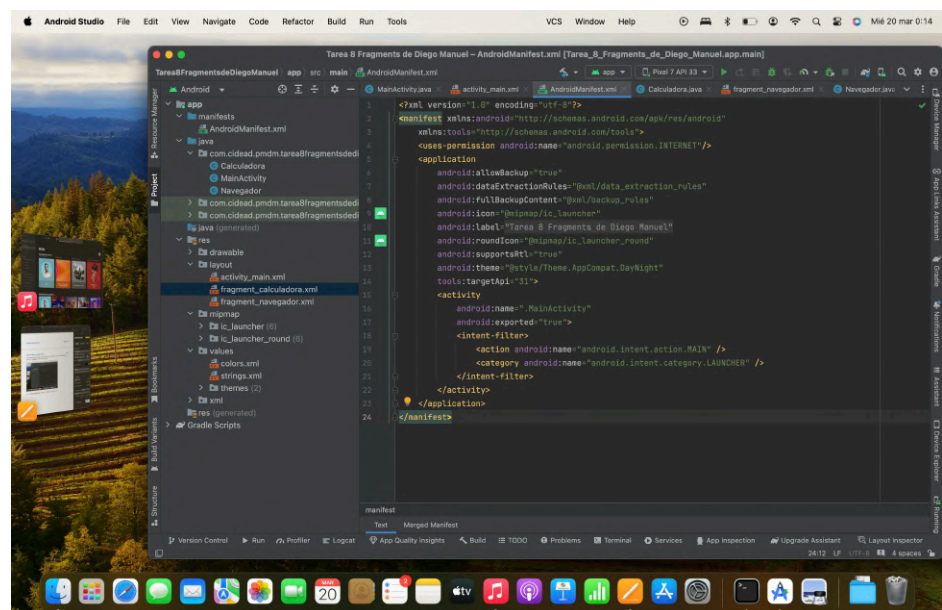
PMDM, debería nombrar esta tarea como...

sanchez\_manas\_begona\_PMDM08\_Tarea

Lo primero que haremos será crear el proyecto al que llamaremos *Tarea 8 Fragments de Diego Manuel*



En el *AndroidManifest* vamos a cambiar el tema a *AppCompatActivity*



A continuación pasaremos a crear y construir el layout que contendrá la calculadora al que llamaremos *fragment\_calculadora*.

Estará compuesto por dos *EditText*, un *RadioGroup* que estará compuesto por cuatro *RadioButton*, un botón y un *TextView*.

Los dos *EditText* serán los encargados de recoger los números introducidos por el usuario para hacer las operaciones matemáticas pertinentes.

Los *RadioButton* serán los encargados de establecer la operación a realizar (sumar, restar, multiplicar y dividir) y al estar

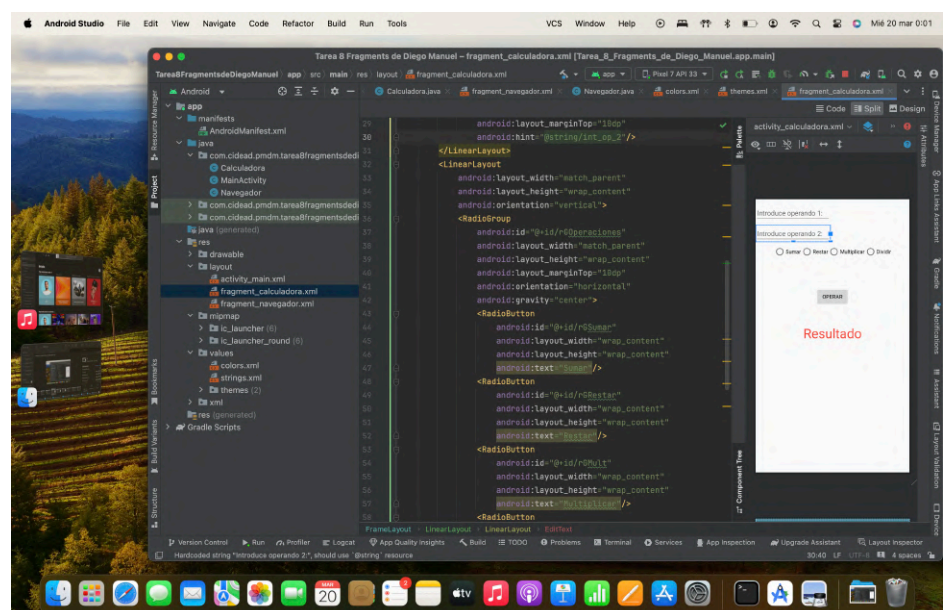
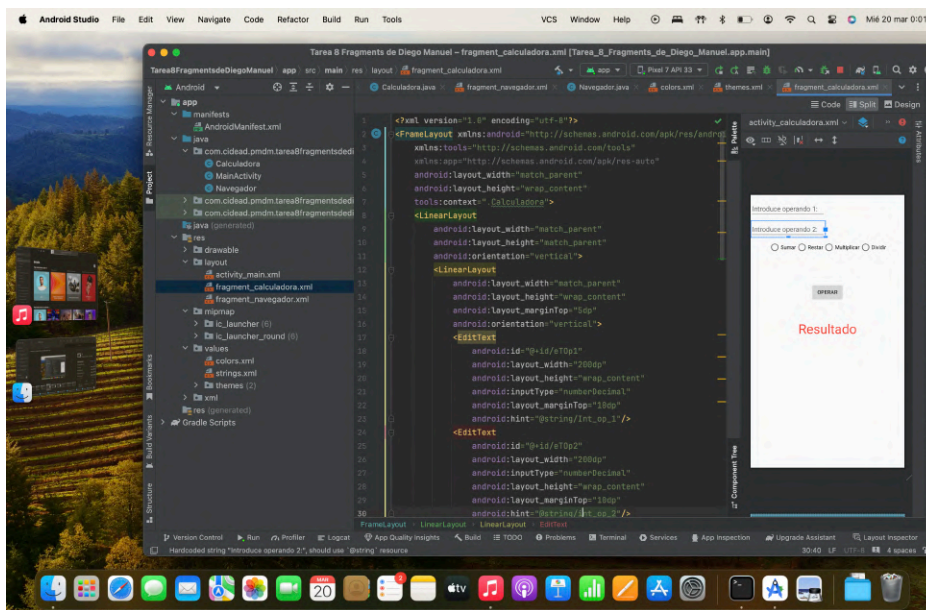


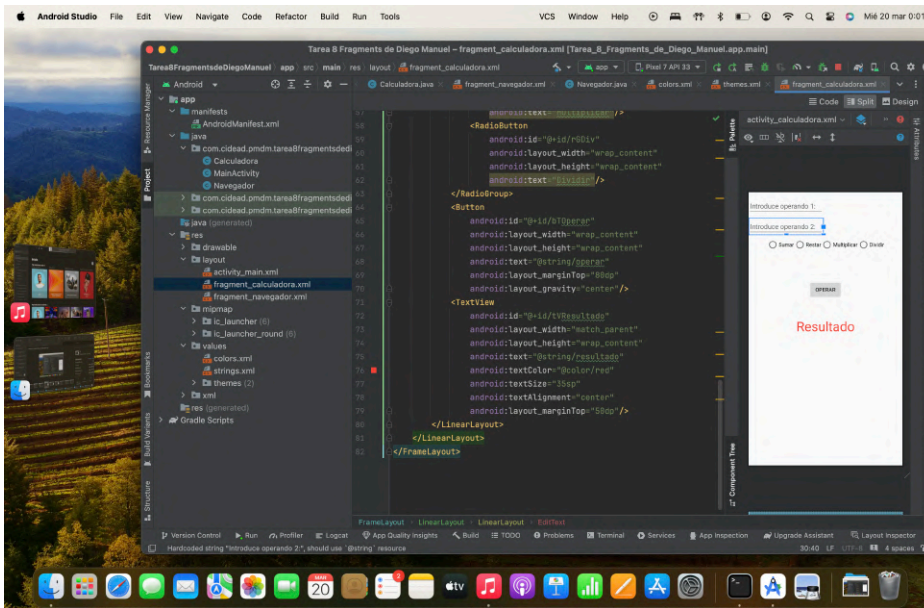
encapsulados en un `RadioGroup` solo puede estar seleccionado uno al mismo tiempo.

El botón será el encargado de lanzar el código para realizar la operación seleccionada por el cliente.

Y por último, en el `TextView` se mostrara el resultado de la operación, o los mensajes del control de exacciones (como dividir entre 0 o dejar vacío algún campo de los `EditText`).

En las siguientes imágenes podemos ver el código de este fichero.





Tras ello vamos a componer el código de la clase *Calculadora*.

Dicha clase extenderá de *fragment* y dispondrá de dos atributos privados, estáticos y finales para contener los parámetros que va a recibir, que para este caso son dos.

También contendrá otros dos atributos de tipo *string* y privados para almacenar los parámetros y que usaremos para el método *onCreate*.

También dispondrá de un método constructor que recibirá los dos parámetros, creará un nuevo *fragment* de la clase *calculadora*, enviara dichos parámetros mediante los método *putString*, establecerá los argumentos mediante el método *setArguments* y retornará el *fragment*.

A continuación contendrá el método *onCreate*.

Dicho método llamará a su método padre y comprobará si ha recibido los argumentos, y si los ha recibido, los establecerá en la variables creadas para tal fin (*mParam1* y *mParam2*).

Lo siguiente que haremos era componer el método *onCreateView*.

Dicho método contendrá los atributos que contiene el *xml* asociado a dicha clase (dos *EditText*, cuatro *RadioButton*, un botón y un *TextView*).

Además contendrá un *View* que será el encargado de inflar el *fragment* en el que no encontramos, *calculadora* en este caso.



Haremos referencia a los objetos de su xml, y llamaremos al método `setOnClickListener` asociado al botón que realizará las operaciones matemáticas.

Dentro de dicho método crearemos el código encargado de realizar las operaciones matemáticas.

Dicho método contendrá tres atributos de tipo `double` (el recogido de cada uno de los `EditText` y el que almacenará el resultado de la operación), convertirá a `string` los `double` recogidos de los `EditText`.

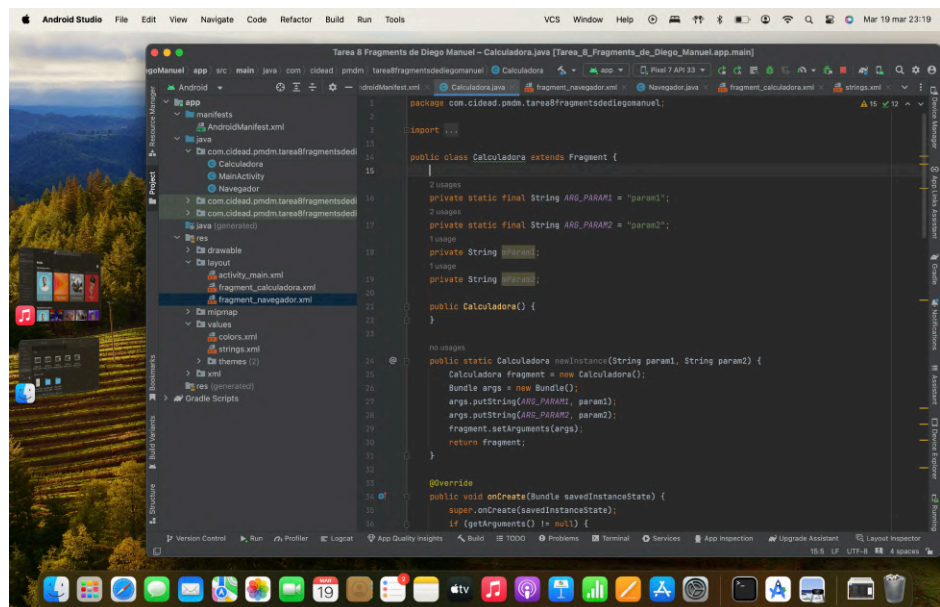
Tras ello comprobará si alguno de los dos `string` está vacío, y en el caso de estar alguno de ellos vacío mostrara en el `TextView` que el mensaje pertinente indicando que no puede estar vacío.

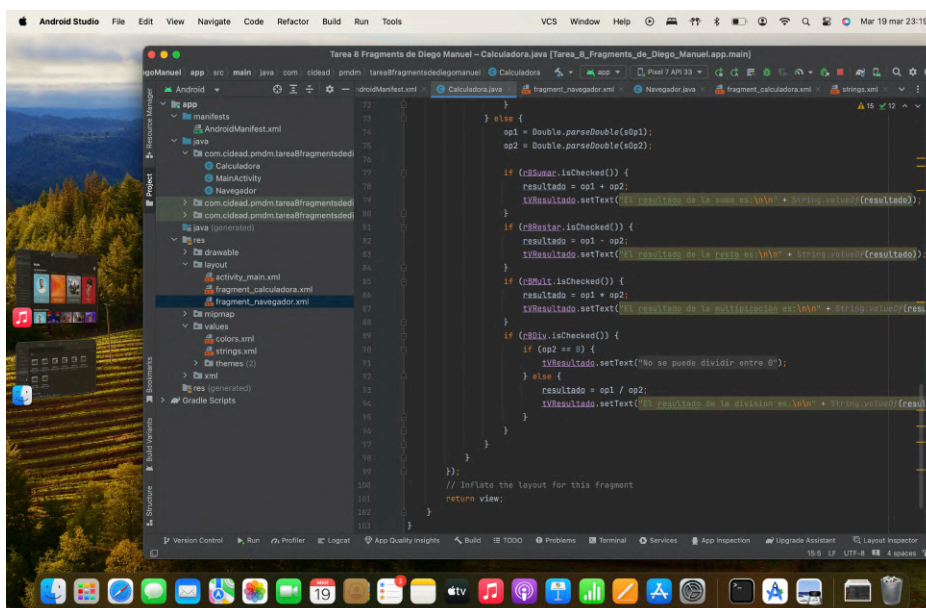
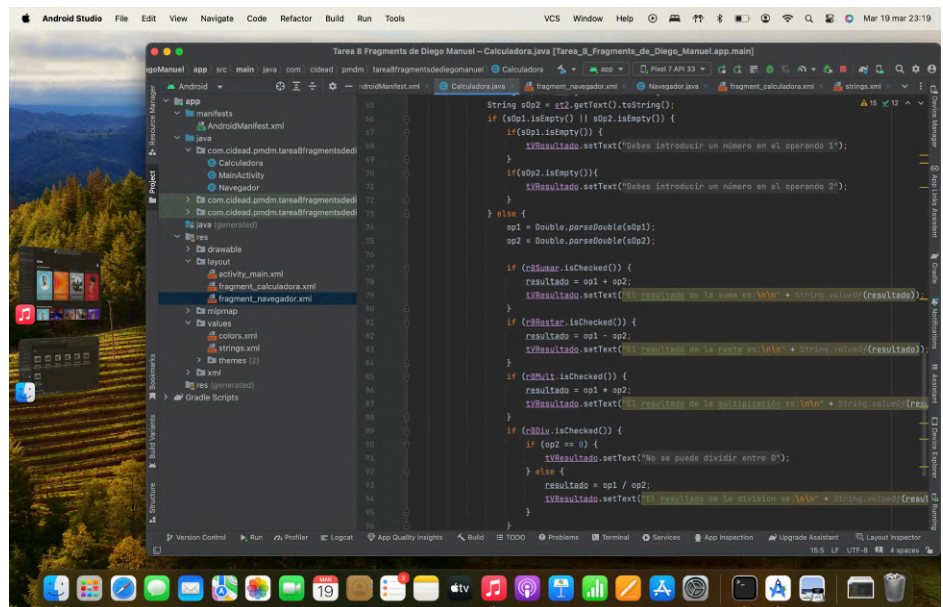
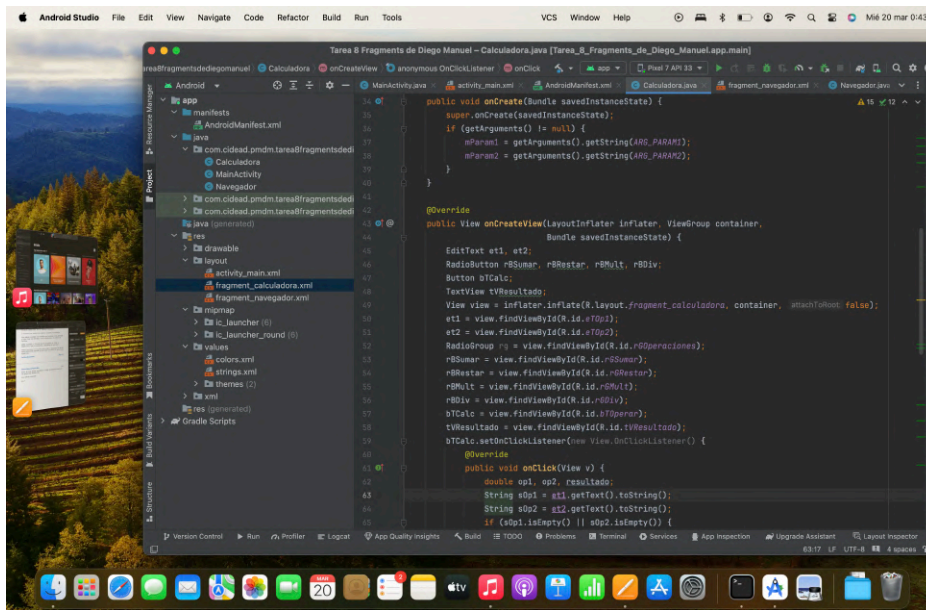
En caso de no estar vacío comprobara cual `RadioButton` está seleccionado para realizar la operación asociada a cada botón.

En el caso de estar seleccionado el `RadioButton` de dividir, primero comprobará que el divisor (operando 2) no es 0, en cuyo caso mostrara el mensaje indicando que no es posible dividir entre 0, y en caso de no ser 0 realizará la operación de división.

Por último retornará el `fragment`.

En las siguientes imágenes podemos ver el código que compone dicha clase.



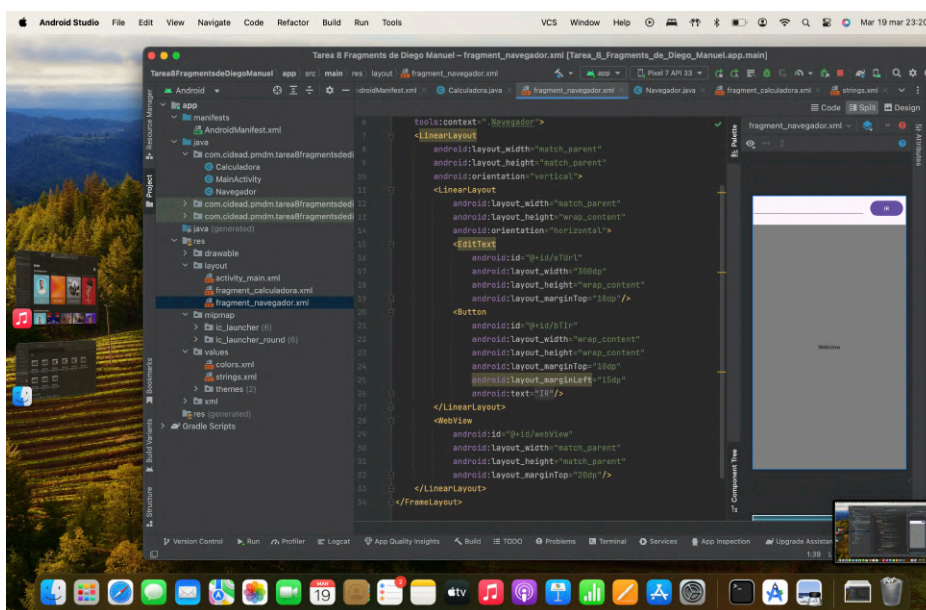
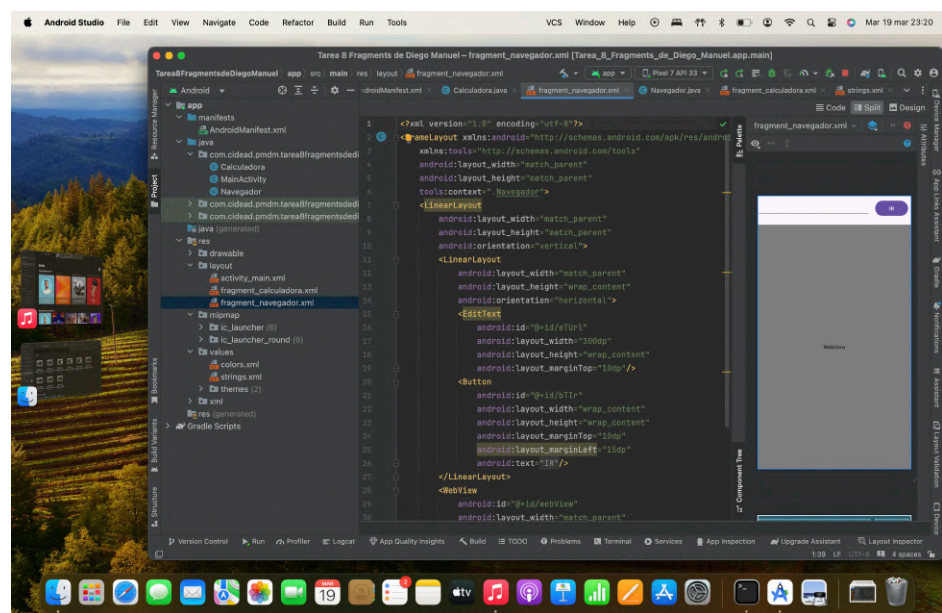


Lo siguiente que haremos será crear y construir el layout y la clase que contendrá el WebView, al que llamaremos `fragment_navegador`.

Estará compuesto por un `EditText` que será el encargado de recoger la dirección introducida por el usuario, un botón que contendrá el código para ejecutar el `WebView` y mostrar en él la web introducida por el usuario y un `WebView` que mostrara dicha web.

Todo esto estará contenido en un `LinearLayout` que a su vez contendrá otro `LinearLayout` para mostrar correctamente el `EditText` y el botón.

En la siguiente imagen podemos ver el código de dicho layout.





A continuación crearemos y construiremos el código de la clase Navegador.

Dicho clase contendrá dos atributos de tipo privados, estáticos y finales que serán los encargados de recibir y almacenar los argumentos (dos en este caso).

Además contendrá otros dos atributos string de tipo privados que murásemos posteriormente para el método onCreate.

A continuación crearemos el constructor por defecto, y el constructor con parámetros, que recibirá los dos parámetros, creará un nuevo fragment de la clase navegador, enviara dichos parámetros mediante los método putString, establecerá los argumentos mediante el método setArguments y retornará el fragment.

Tras ello compondremos el método onCreate.

Dicho método llamará a su método padre y comprobará si ha recibido los argumentos, y si los ha recibido, los establecerá en la variables creadas para tal fin (mParam1 y mParam2).

Posteriormente compondremos el método onCreateView.

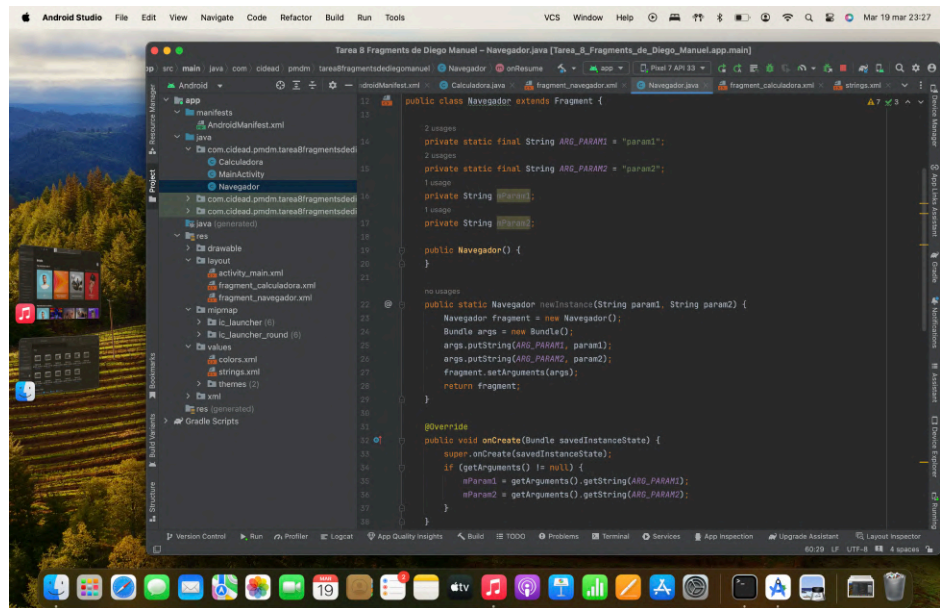
Dicho método contendrá los atributos que contiene el xml asociado a dicha clase (un EditText, un botón y un WebView).

Además contendrá un View que será el encargado de inflar el fragment en el que no encontraos, navegador en este caso.

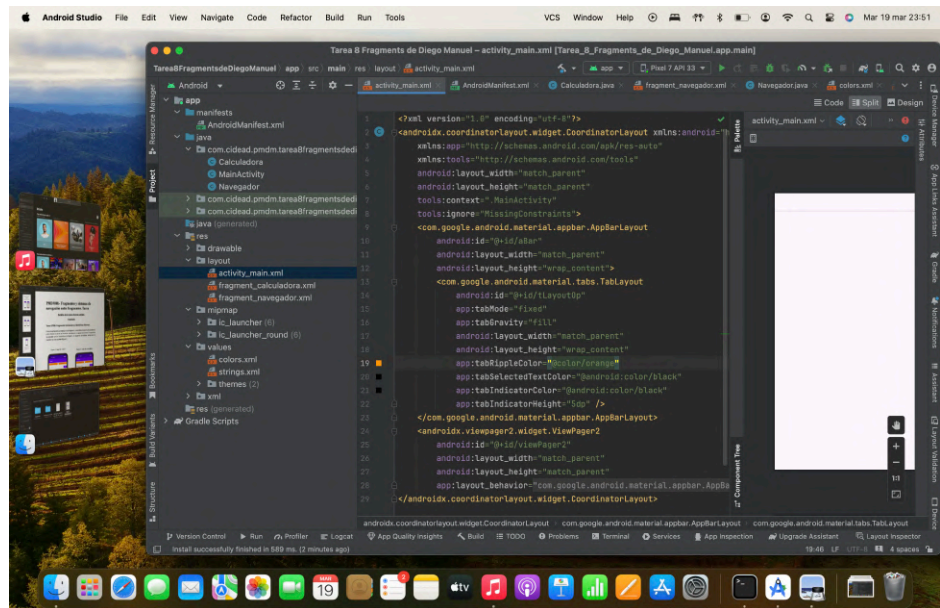
Haremos referencia a los objetos de su xml, y llamaremos al método setOnClick asociado al botón que se encargará de recoger la dirección web introducida por el usuario y concatenarla con el string "https://" para que no nos de error el permiso de internet.

Por último retornará el fragment.

En las siguientes imágenes podemos ver el código de dicha clase.







Por último vamos a componer el código de la clase MainActivity.

Dicha clase contendrá los mismos atributos que su xml asociado (un ViewPager2) y un FragmentStateAdapter.

En el método onCreate haremos referencia a los atributos asociado en su xml y crearemos un nuevo pageAdapter, un TabLayout.

En el método onConfigureTab le pasaremos como parámetros el tab y la posición.

Dentro de dicho método crearemos un switch, que en el case 0 estableciera el texto calculadora y en el case 1 el texto navegador.

La posición 1 será la que se muestre por defecto, por ello lo indicaremos en el método setCurrentItem.

Tras ello crearemos el método ScreenSlidePagerAdapter que extenderá de FragmentStateAdapter que llamará a su método padre.

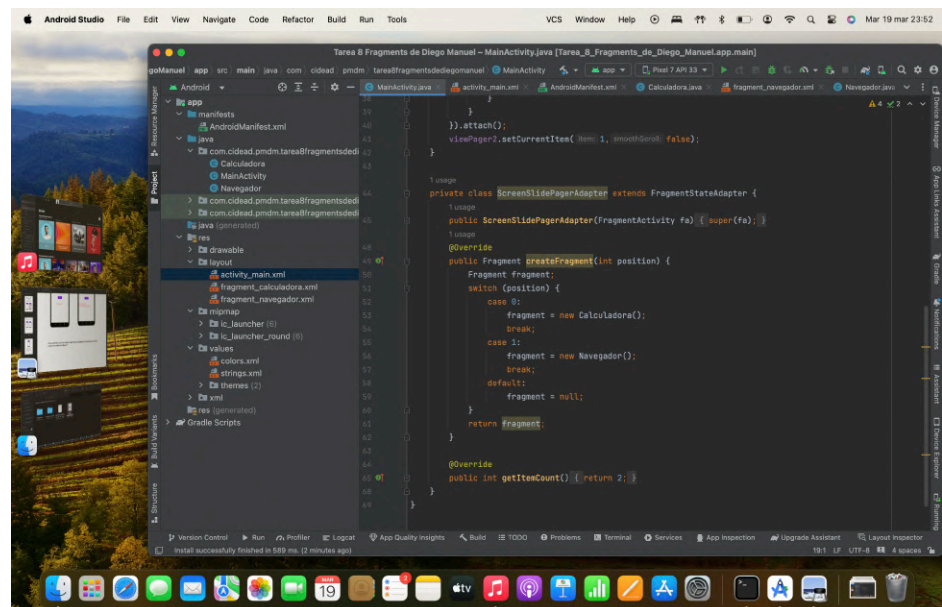
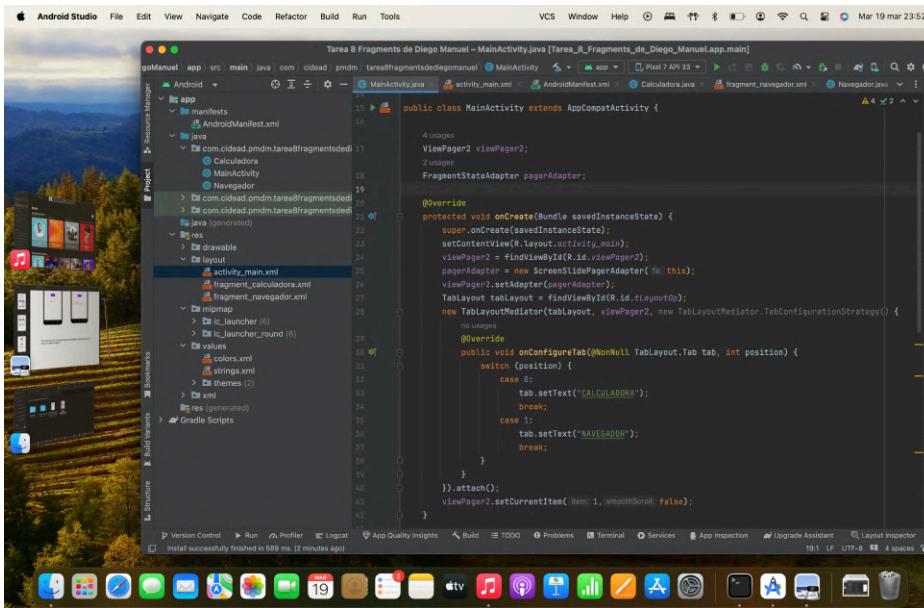
A continuación sobre escribiremos el método createFragment, al cual le pasaremos por parámetro la posición (es decir, el fragment seleccionado).

Crearemos un Fragment y tras ello un switch que recibirá la posición.

En el case 0 iniciará el fragment de la calculadora, en el case 1 el del navegador, en default instanciará a null el fragment y retornará el fragment pertinente.

Por último sobre escribiremos el método getItemCount, al cual le diremos que el número de item son 2.

En las siguientes imágenes podemos ver el código de dicha clase.



Las capturas de el inicio de la aplicación con una web que empiece por la inicial de mi nombre y una captura de la calculadora haciendo una operación son las imágenes presentes en la portada.

Y con esto damos por finalizada esta tarea.