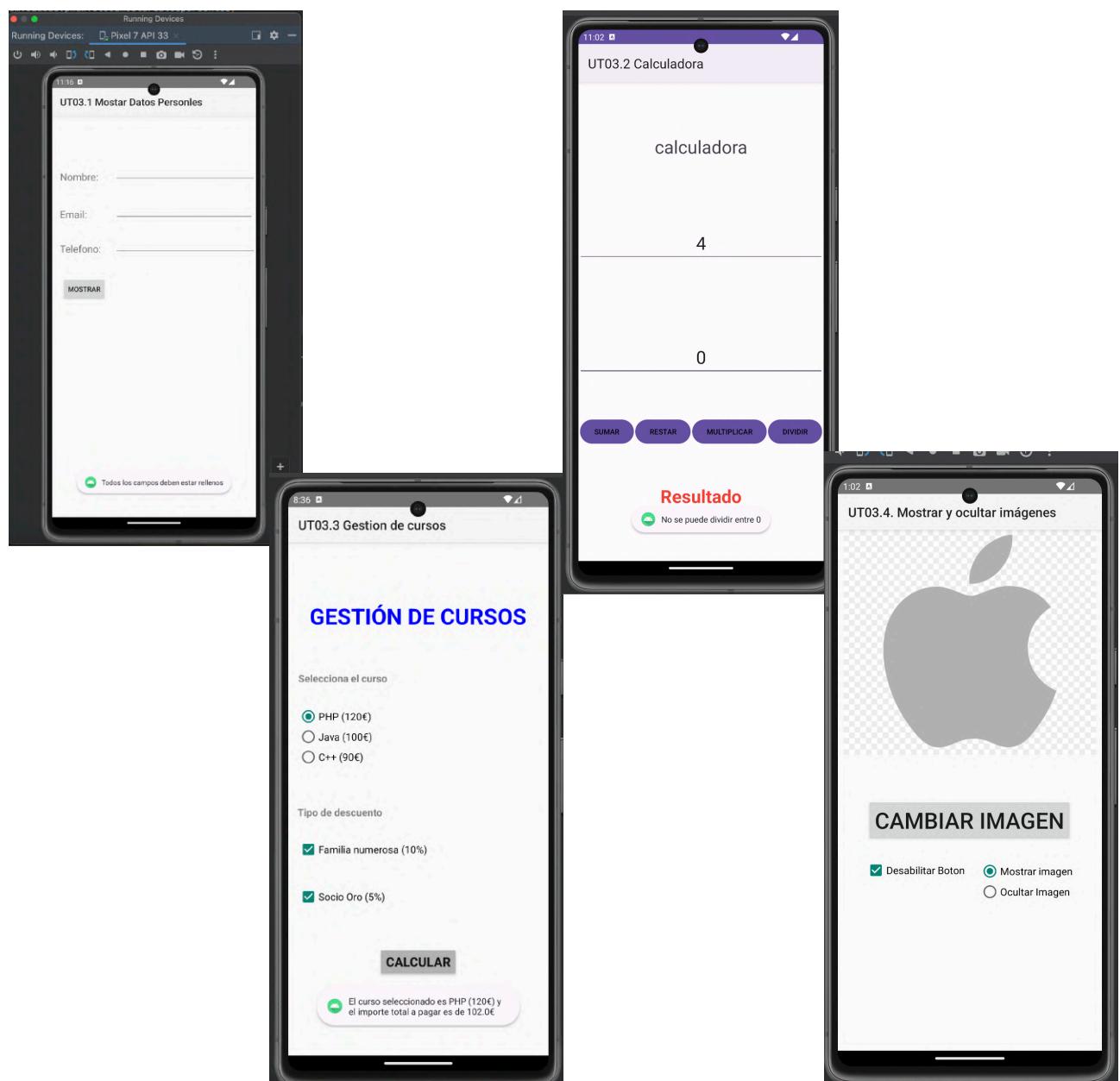


# Tarea 3 de Programación Multimedia y Dispositivos Móviles



Diego Manuel Carrasco Castañares

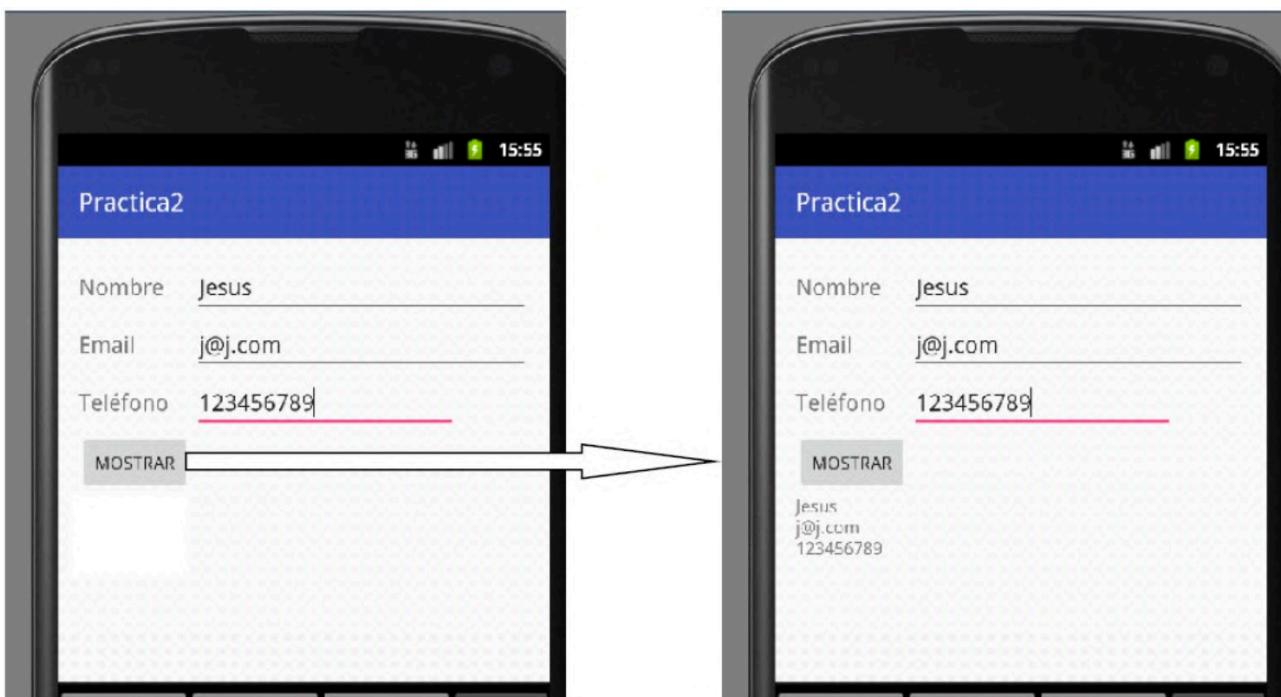
Tarea PMDM03.- Conceptos básicos para la programación de aplicaciones Android.

Detalles de la tarea de esta unidad.

Enunciado:

**Tarea UT03.1. Mostrar datos personales.**

Vamos a realizar una pequeña app que pida unos datos y cuando se pulse en el botón Mostrar, se muestren los datos introducidos en una etiqueta:



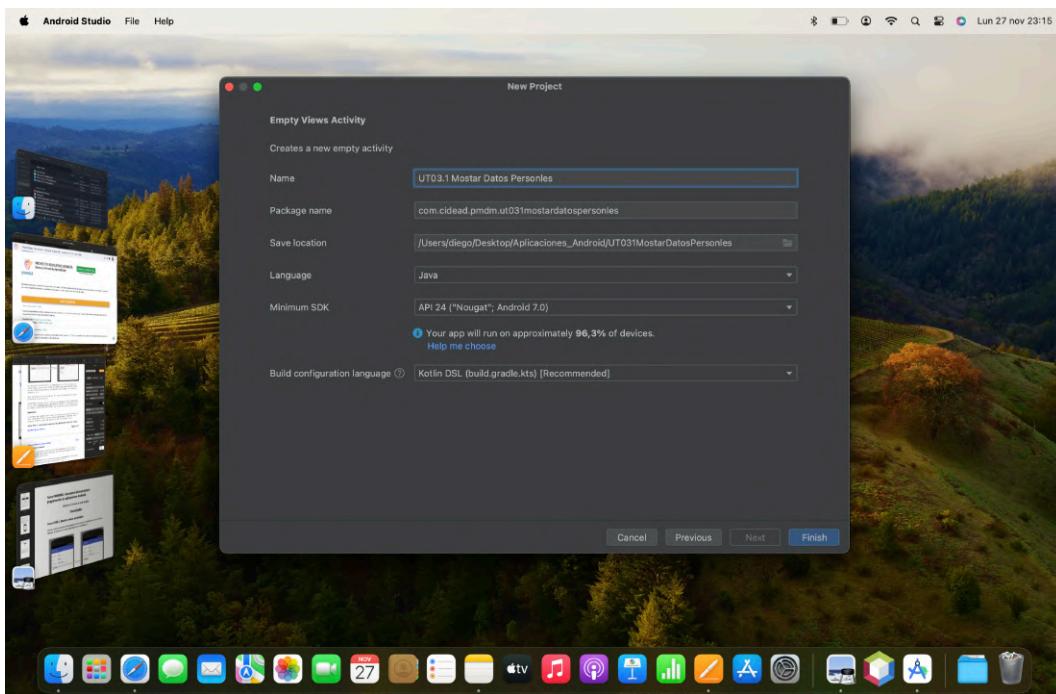
El resultado se mostrará en un TextView con un tipo de letra de tamaño 30sp y color azul. Tener en cuenta que cada EditText debe ser del tipo de dato que se introduzca, por ejemplo, contraseña, numérico, etc.

Para resolverlo utilizar captura del evento programándolo cuando se pulsa el botón mostrar.

**IMPORTANTE:** Entrega todo el proyecto de android studio comprimido. En el archivo MainActivity.java deberá aparecer un comentario con el nombre y apellidos del alumno que ha desarrollado el proyecto.

Desarrollo:

Lo primero que haremos será crear un proyecto en Android Studio al cual llamaremos "UT03.1 Mostar Datos personales". Elegimos como lenguaje "Java" y como SDK mínimo el API 24 (Android 7,0).

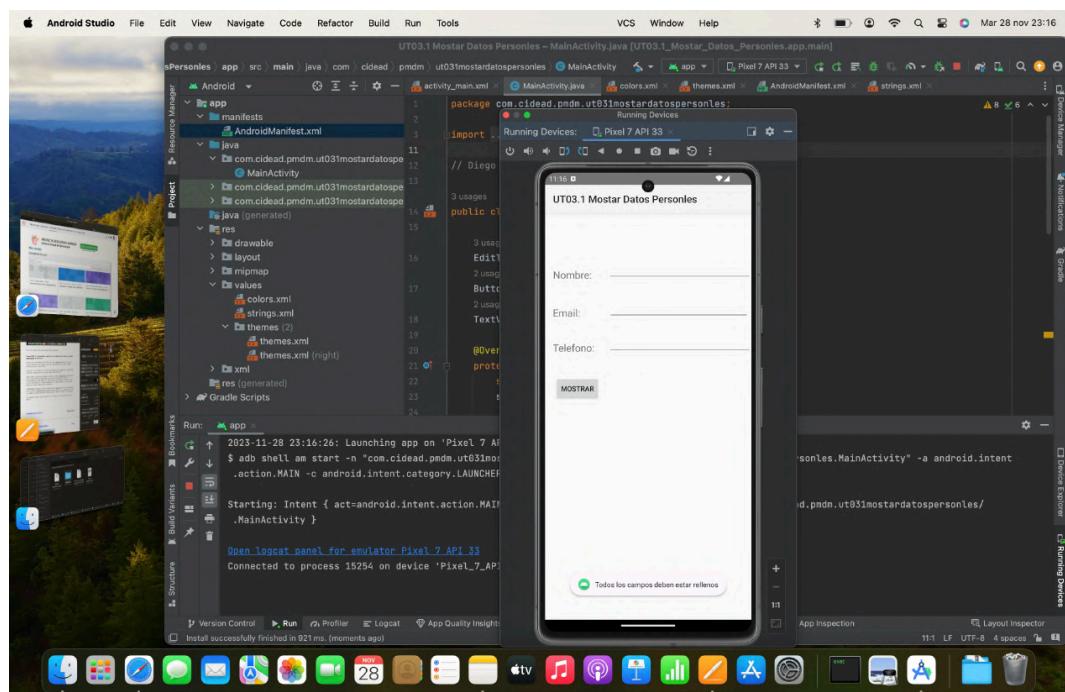


Tras ello vamos a crear los tres textView llamados nombre, email y teléfono respectivamente, los tres EditText para introducir los datos y por último el button.

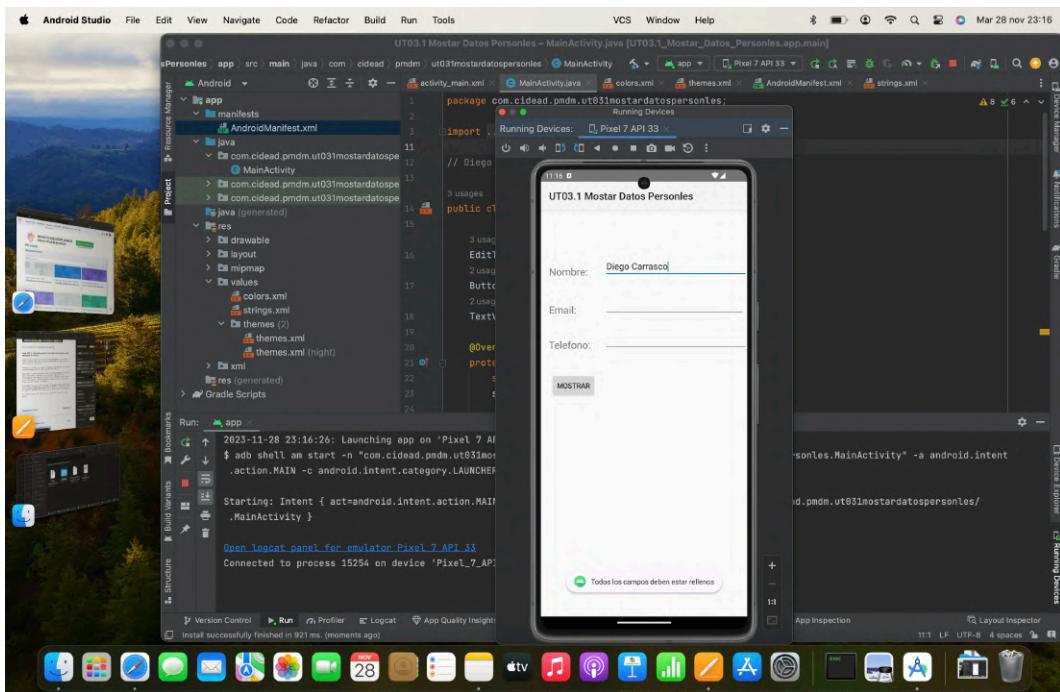
Lo siguiente que haremos será capturar el evento al clicar el botón de forma programática.

Por último vamos a usar la clase Toast para mostrar en una ventana emergente si el usuario no introduce todos los datos.

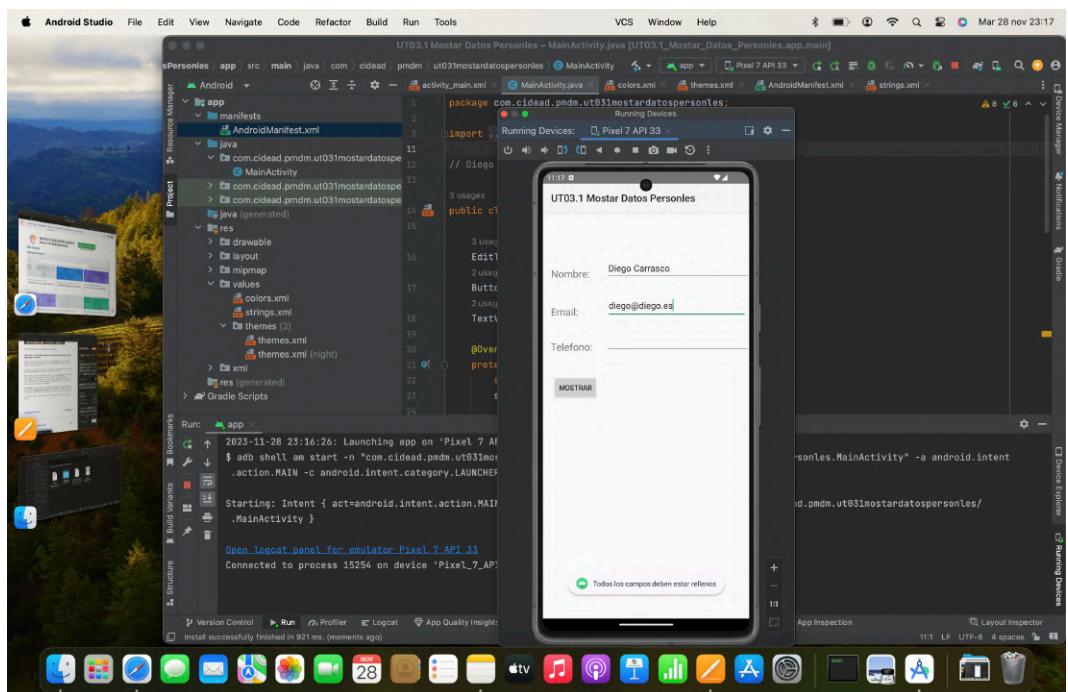
En las siguientes imágenes podemos verlo. La primera sin ningún campo relleno muestra que se deben llenar todos los datos



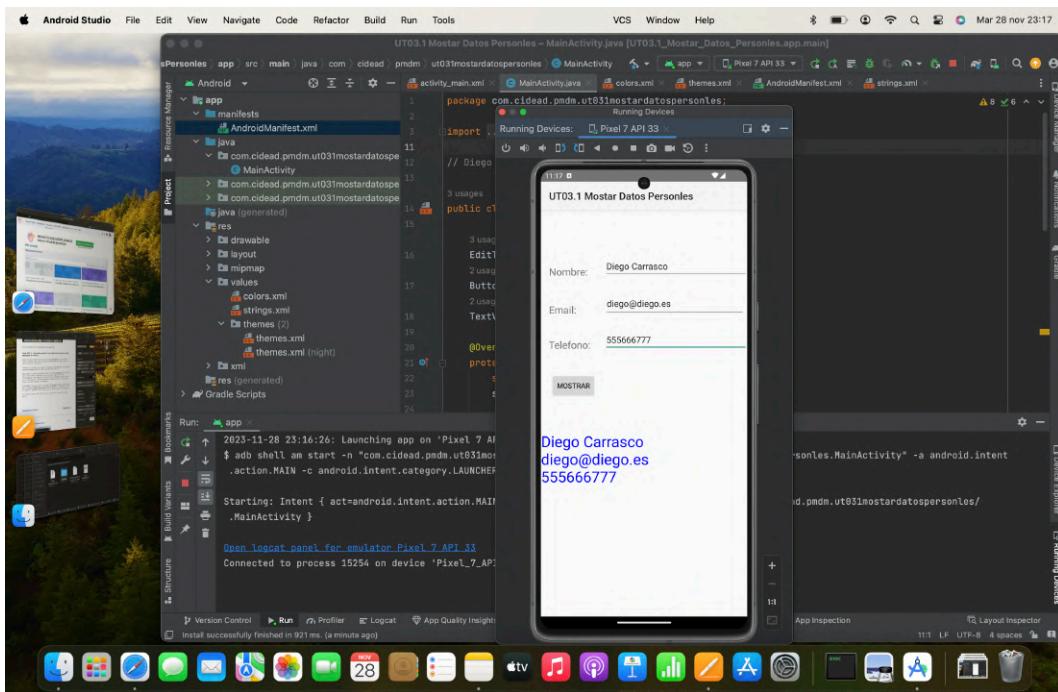
En esta imagen podemos apreciar como solo rellenamos el nombre y muestra el mensaje de que se deben llenar todos los datos sin llegar a mostrar nada en el textview.



En esta imagen podemos apreciar como solo rellenamos el nombre y el email y muestra el mensaje de que se deben llenar todos los datos sin llegar a mostrar nada en el textview.



En esta imagen podemos apreciar como rellenamos todos los datos y muestra en el textview la información escrita y recogida de los edittext.



Con esto damos por finalizada este apartado.

### Tarea UT03.2. Calculadora sencilla con definición única de clase manejadora de eventos.

Hacer una calculadora sencilla con dos operandos de tipo real y cuatro botones para poder realizar operaciones de suma, resta, multiplicación y división.

Capturar los eventos de clicado de manera programática. Controlar posibles excepciones de tipo y de división por 0.

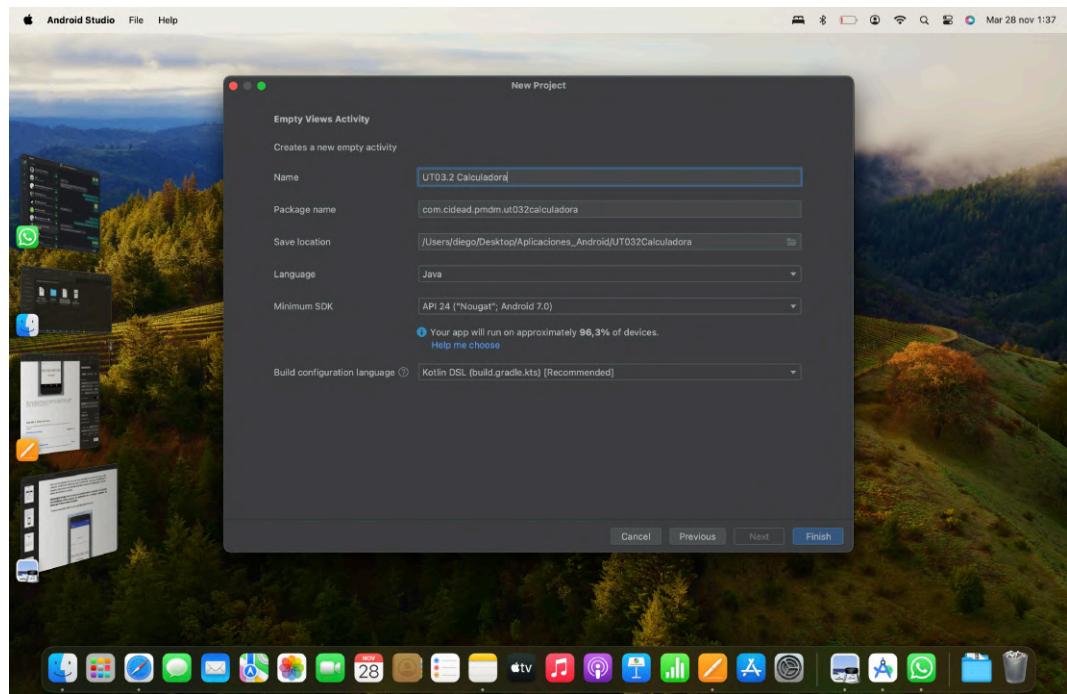
Debe tener la particularidad de que la clase manejadora de evento de clicado sobre cualquiera de los botones debe ser única. Internamente, será capaz de identificar el elemento sobre el que se ha realizado el clicado mediante el método getId () aplicado sobre la vista (botón) en cuestión.

**IMPORTANTE:** Entrega todo el proyecto de android studio comprimido. En el archivo MainActivity.java deberá aparecer un comentario con el nombre y apellidos del alumno que ha desarrollado el proyecto.

El aspecto visual será similar a la captura de la pantalla del móvil:



Para esta tarea crearemos un proyecto en Android Studio llamado UT03.2 Calculadora. Elegimos "Java" como lenguaje y como SDK mínimo el API 24 (Android 7,0).



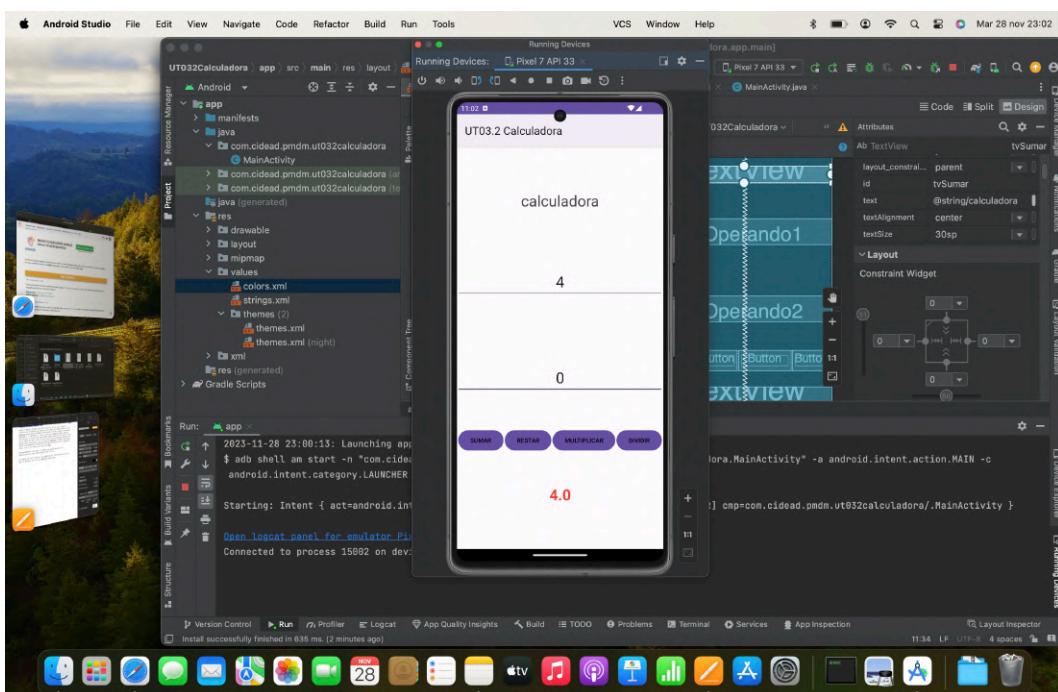
Vamos a añadir primero la parte gráfica, en la cual vamos a incluir un textview que mostrará el texto de "Calculadora", dos edittext para que el usuario introduzca los dos números llamados operando 1 y operando 2, cuatro botones llamados sumar, restar, multiplicar y dividir de los cuales capturaremos sus eventos (clic en el botón en este caso) en una única clase para que realice las operaciones oportunas y por último un textview llamado resultado donde se mostrara el resultado de cada operación con el texto en color rojo y un tamaño de letra de 30sp.

Posteriormente crearemos el código para que realice las operaciones de cada botón dentro del onClick.

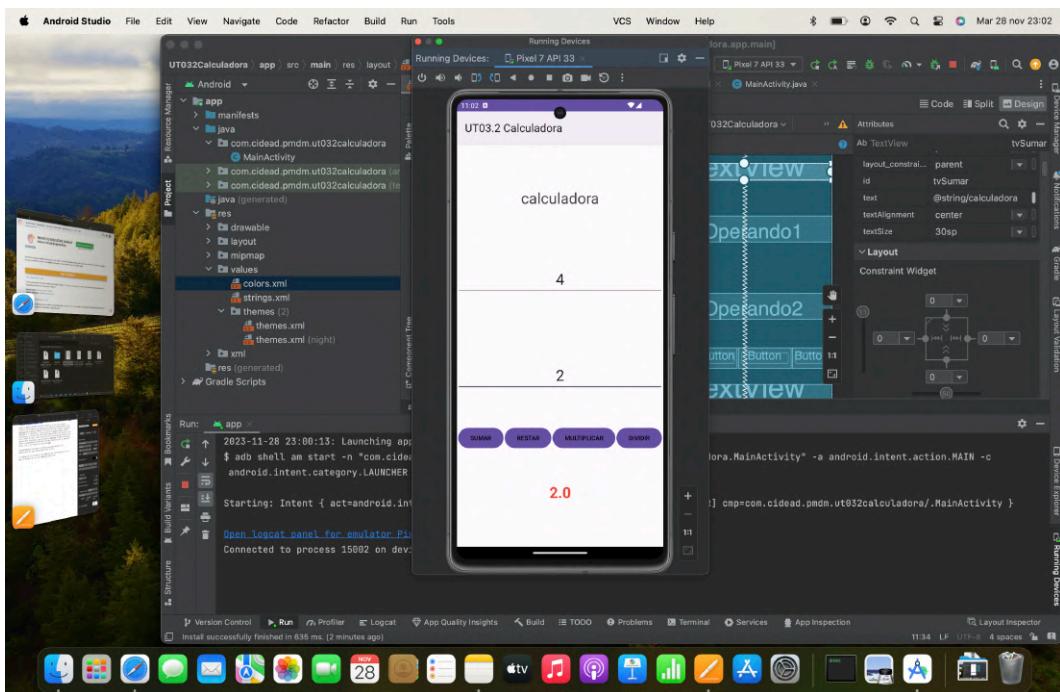
Para controlar la división entre 0 lo que hacemos es mostrar una ventaja emergente mediante la clase Toast indicando que no podemos dividir entre 0.

El código de los ficheros y la captura del funcionamiento de la app lo podemos ver en las imágenes siguientes.

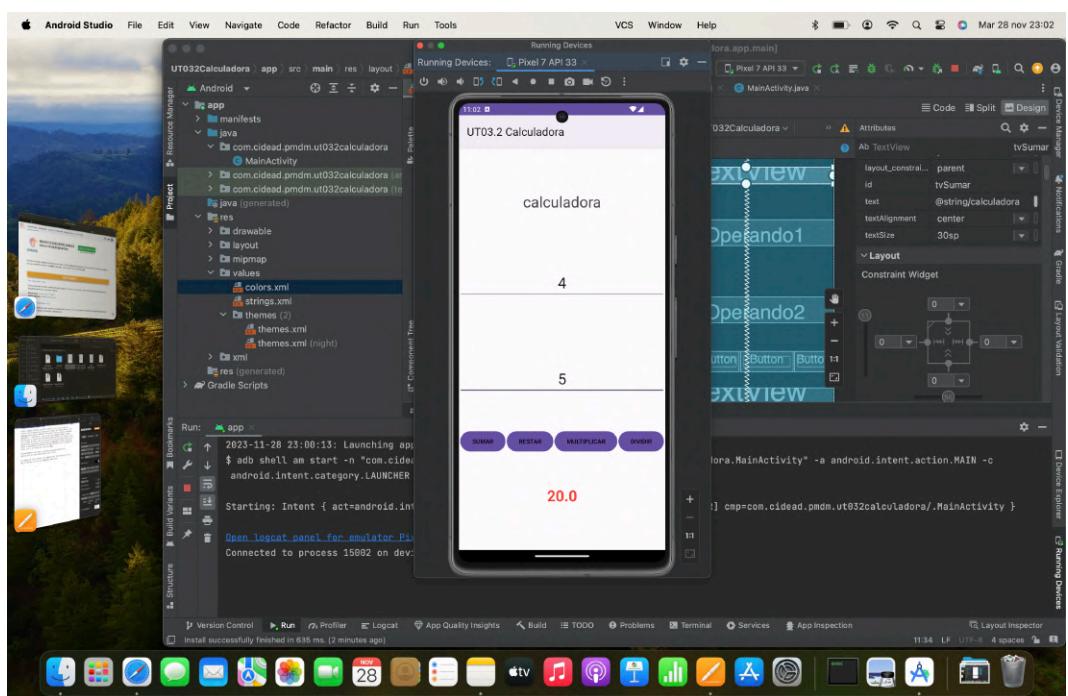
Suma completada:



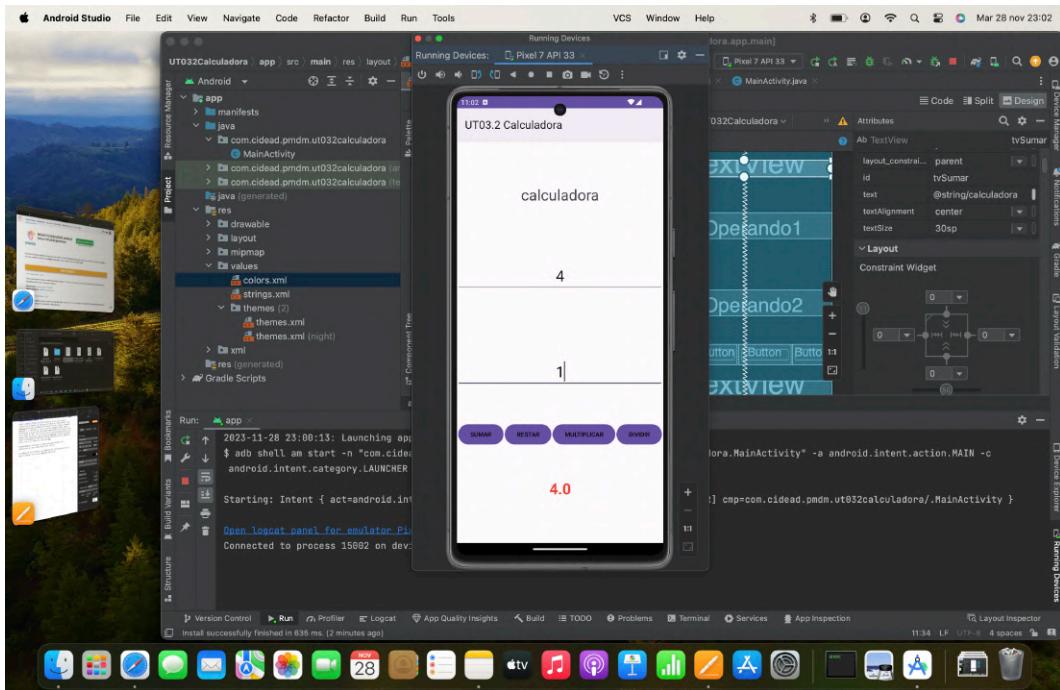
Resta completada:



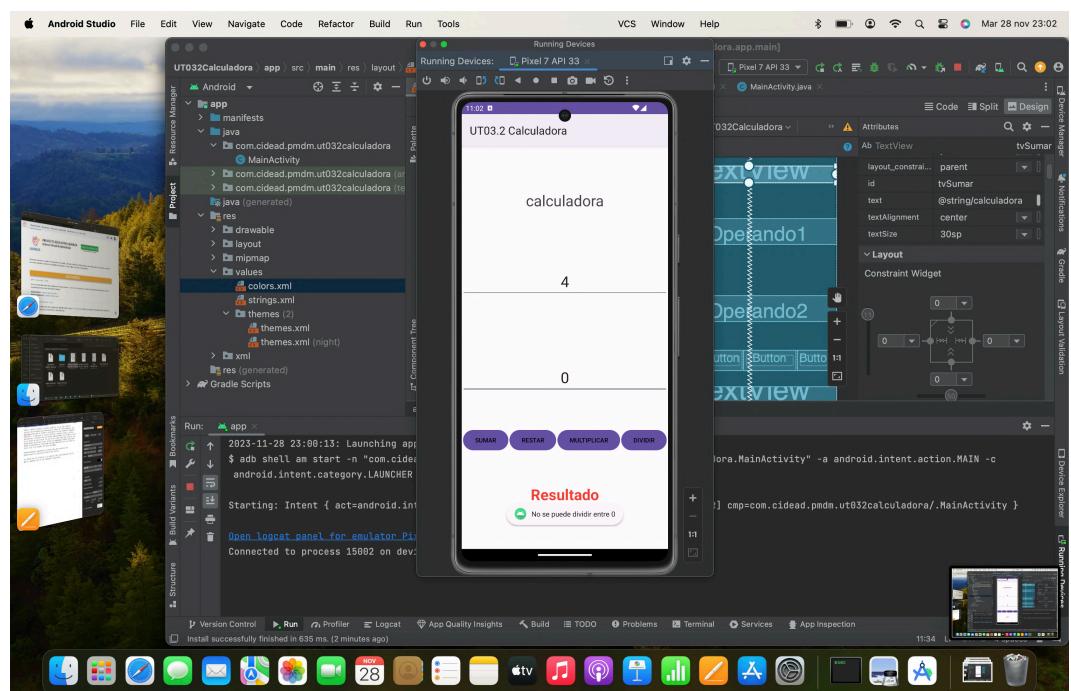
Multiplicación completada:



Division normal completada:



Division entre 0 completada:



En las dos imágenes siguientes mostramos el código del MainActiviti.java.

```

UT03_2 Calculadora - MainActivity.java [UT03_2_Calculadora.app.main]

package com.cidead.pmdm.ut032calculadora;

import ...

// Diego Manuel Carrasco Castañares
public class MainActivity extends AppCompatActivity implements View.OnClickListener{

    TextView tvResultado;
    EditText etOperando1;
    EditText etOperando2;
    Button btSumar, btRestar, btMulti, btDiv;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        tvResultado = (TextView) findViewById(R.id.tvResultado);
        etOperando1 = (EditText) findViewById(R.id.etOperando1);
        etOperando2 = (EditText) findViewById(R.id.etOperando2);
        btSumar = (Button) findViewById(R.id.btSumar);
        btSumar.setOnClickListener(this);
        btRestar = (Button) findViewById(R.id.btRestar);
        btRestar.setOnClickListener(this);
        btMulti = (Button) findViewById(R.id.btMulti);
        btMulti.setOnClickListener(this);
        btDiv = (Button) findViewById(R.id.btDiv);
        btDiv.setOnClickListener(this);
    }
}

```

```

@Override
public void onClick(View v) {
    Double operando1 = Double.parseDouble(etOperando1.getText().toString());
    Double operando2 = Double.parseDouble(etOperando2.getText().toString());
    Double dSuma, dResta, dMulti, dDiv;
    String suma, resta, multi, div;
    int id = v.getId();

    if (id == R.id.btSumar){
        dSuma = operando1 + operando2;
        suma = String.valueOf(dSuma);
        tvResultado.setText(suma);
    } else if (id == R.id.btRestar){
        dResta = operando1 - operando2;
        resta = String.valueOf(dResta);
        tvResultado.setText(resta);
    } else if (id == R.id.btMulti){
        dMulti = operando1 * operando2;
        multi = String.valueOf(dMulti);
        tvResultado.setText(multi);
    } else if (id == R.id.btDiv){
        if(operando2 > 0) {
            dDiv = operando1 / operando2;
            div = String.valueOf(dDiv);
            tvResultado.setText(div);
        } else{
            String mensaje = "No se puede dividir entre 0";
            tvResultado.setText("");
            Toast.makeText(context, mensaje, Toast.LENGTH_LONG).show();
        }
    }
}

```

Con esto damos por finalizado este apartado.

**Tarea UT03.3. Gestión de cursos.**

Se va a realizar una app Android como la que se muestra en la imagen.

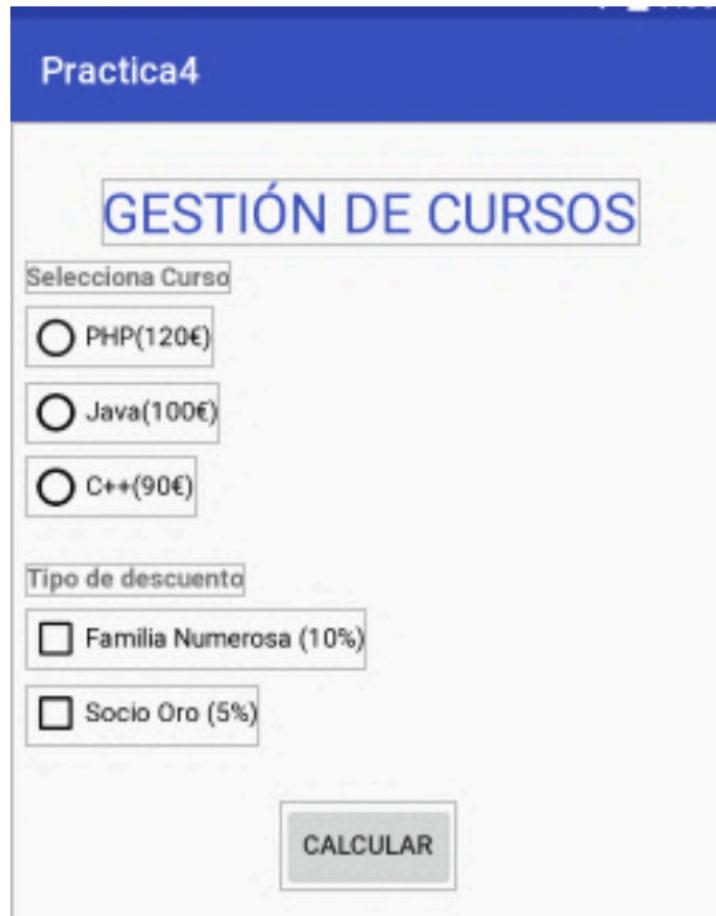
La aplicación calcula el importe de un curso en función de los descuentos marcados.

Los cursos que se representan con RadioButtons deben incluirse dentro de un control llamado RadioGroup para que todos pertenezcan al mismo grupo y solamente se pueda marcar 1.

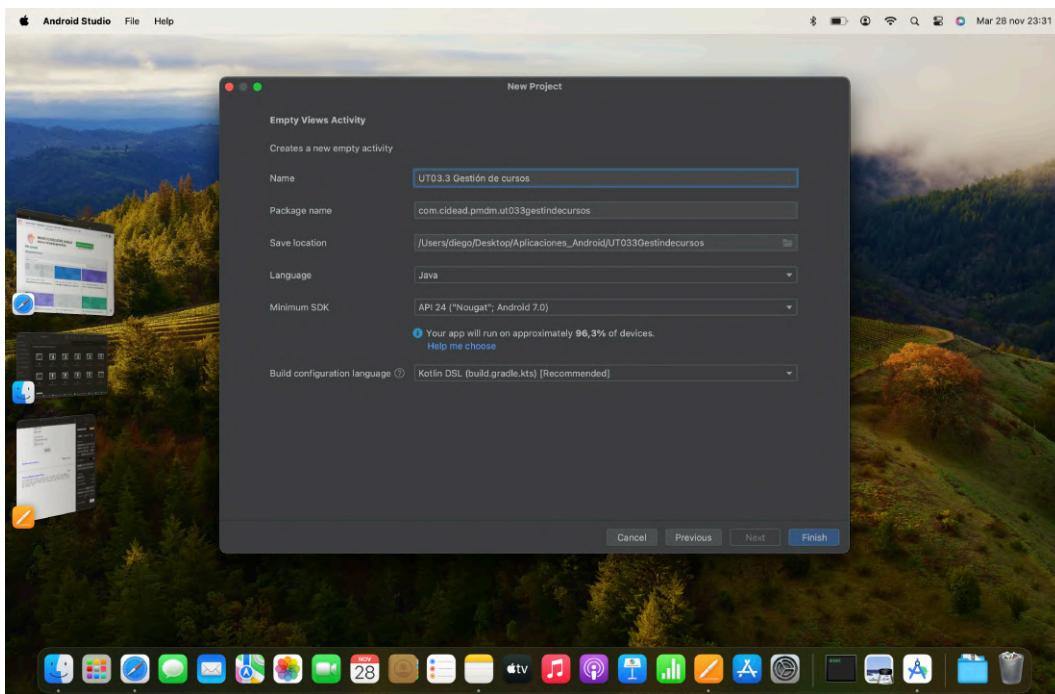
Los descuentos son CheckBox y se pueden seleccionar los dos que están disponibles.

Al dar al botón "Calcular" ,capturando su evento, se debe mostrar a través de un toast un mensaje indicando el curso seleccionado y el importe final después de aplicar los descuentos si han sido seleccionados.

**IMPORTANTE:** Entrega todo el proyecto de android studio comprimido. En el archivo MainActivity.java deberá aparecer un comentario con el nombre y apellidos del alumno que ha desarrollado el proyecto.



Lo primero que haremos en este caso, al igual que en los demás, será el crear el proyecto en Android Studio llamado UT03.3 Gestión de cursos. Elegimos "Java" como lenguaje y como SDK mínimo el API 24 (Android 7,0).



Posteriormente crearemos la parte gráfica para lo cual añadiremos tres textview con los nombre "GESTION DE CURSOS", "Selecciona curso" y "Tipo de descuento" respectivamente.

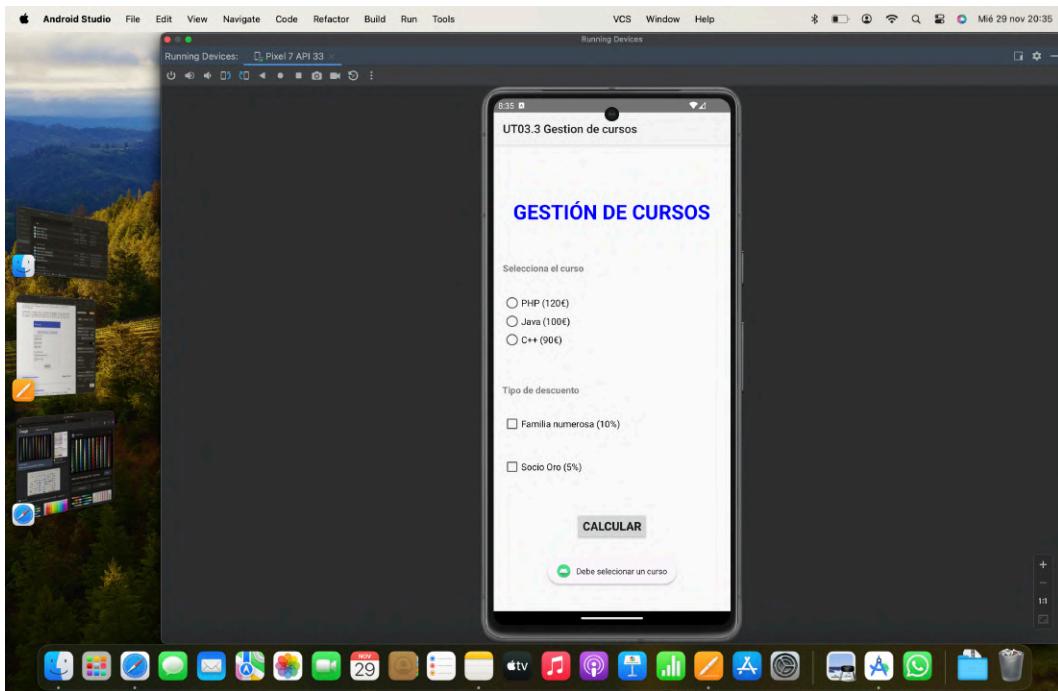
A continuación añadiremos tres radiobuttons a los que llamaremos "PHP (120€)", "Java (100€)" y "C++ (90€)" respectivamente.

Lo siguiente será crear un radiogrup donde agruparemos los tres radiobuttons creados anteriormente.

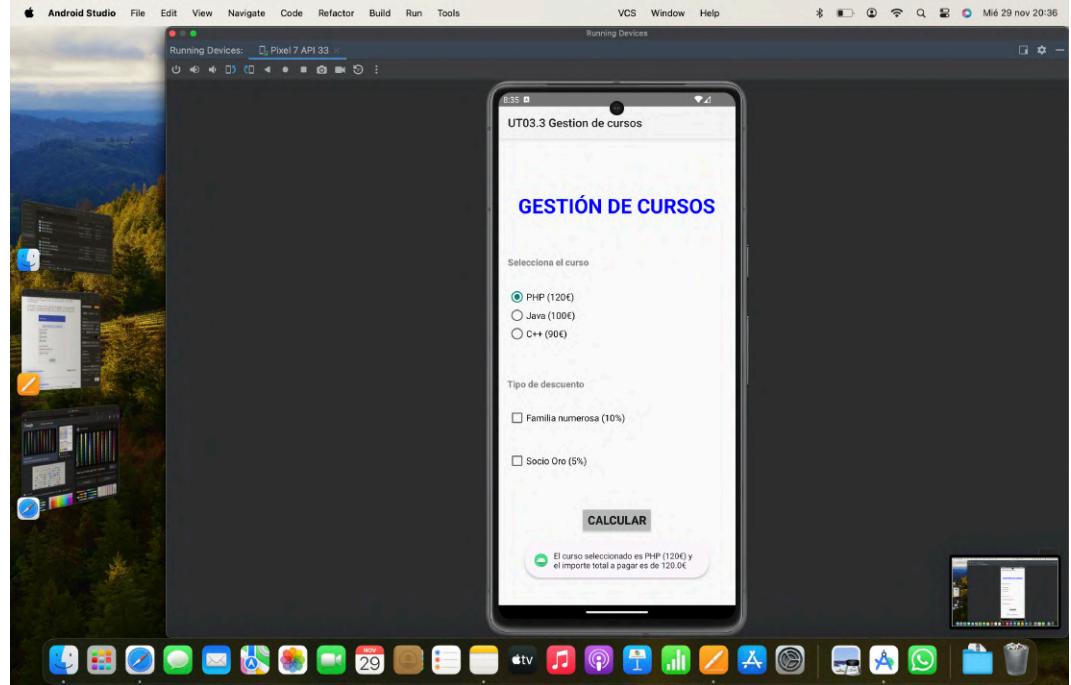
Posteriormente crearemos los dos checkbox, uno llamado "Familia Numerosa (10%) y el otro llamado "Socio Oro (5%).

Por último creamos un button llamado "Calcular"

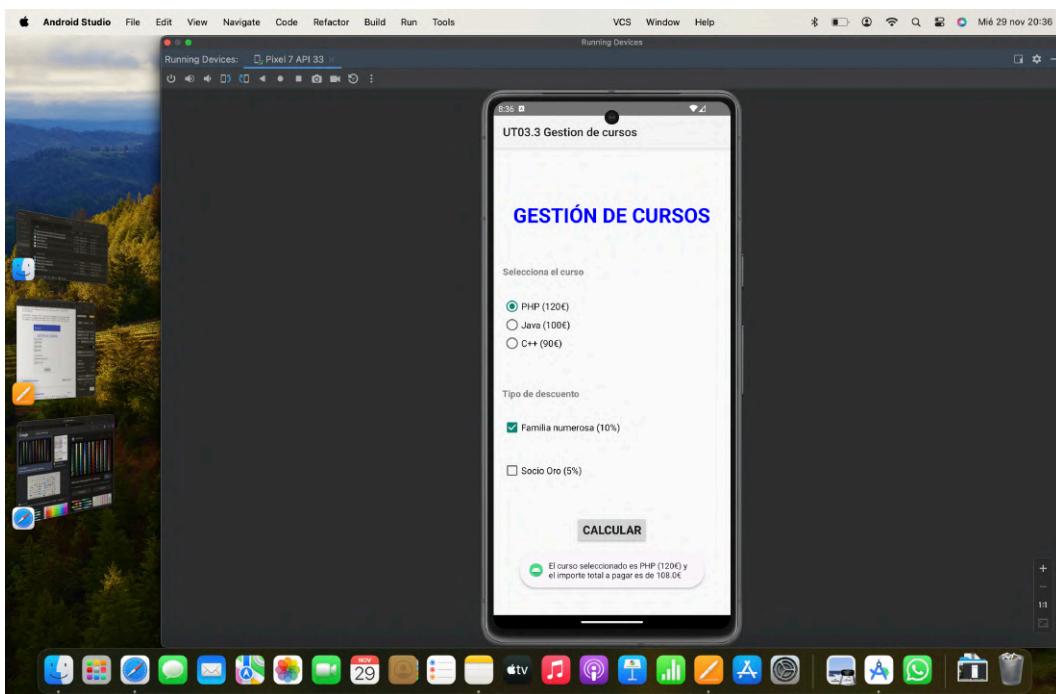
En la siguiente imagen podemos ver como nos muestra el mensaje en un toast "Debe seleccionar un curso" al no seleccionar ningún curso.



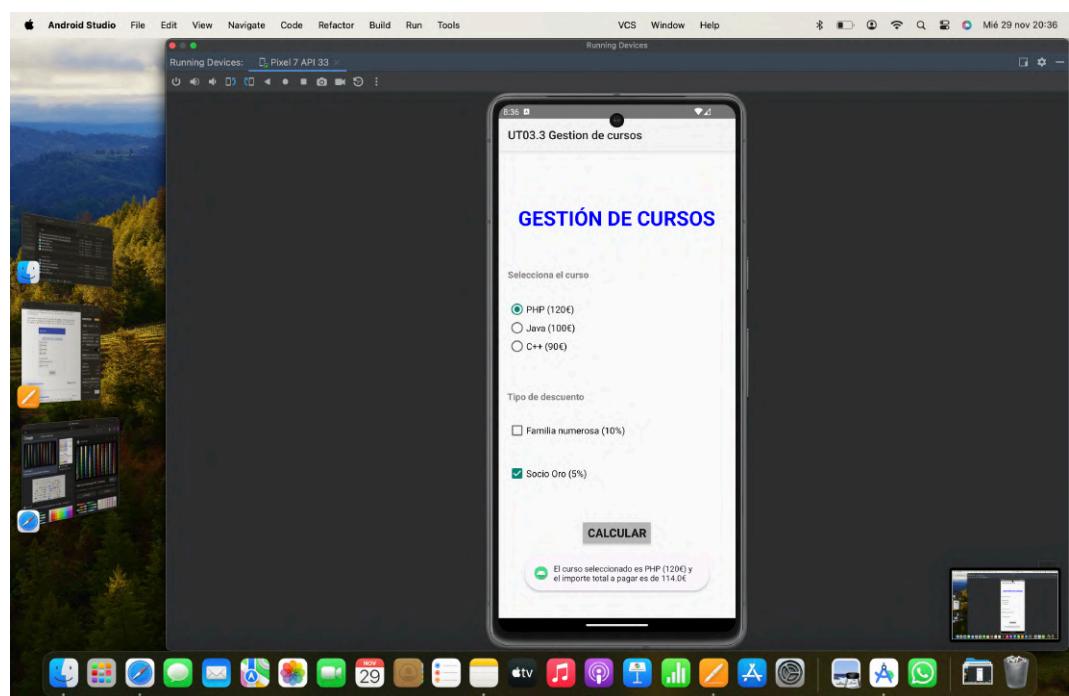
En la siguiente imagen podemos ver como nos muestra el curso seleccionado y el importe a pagar al seleccionar un curso. En nuestro caso, el de PHP.



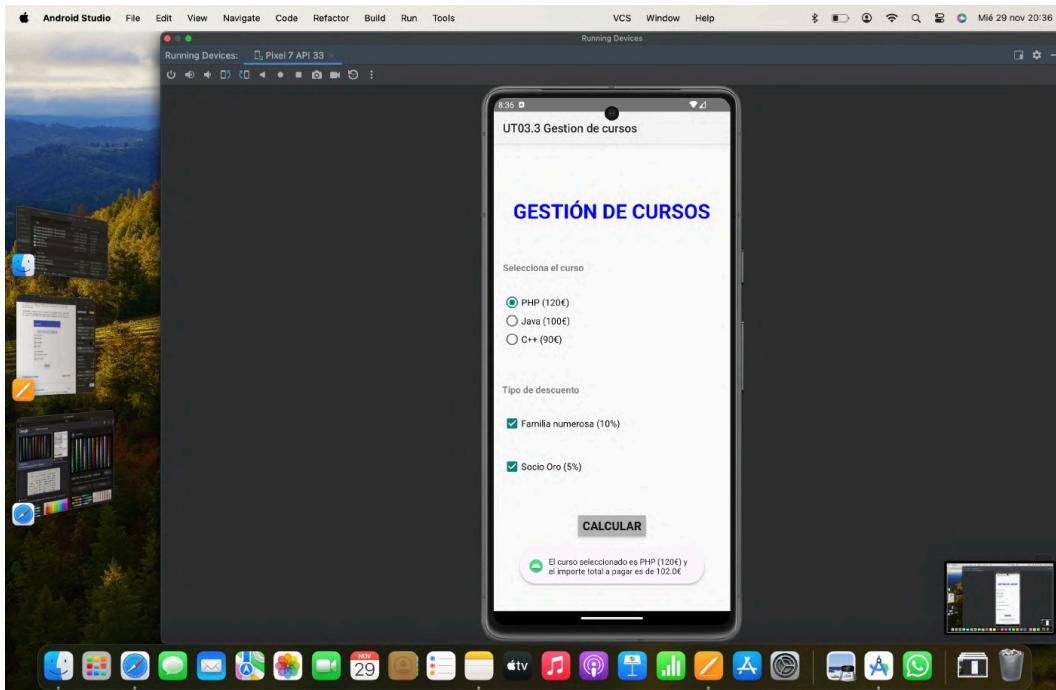
En la siguiente imagen seleccionamos el descuento de familia número al mismo curso de la imagen anterior.



En la siguiente imagen seleccionamos el descuento de socio oro y desmarcamos el descuento de familia numerosa.



En la siguiente imagen seleccionamos ambos descuentos.

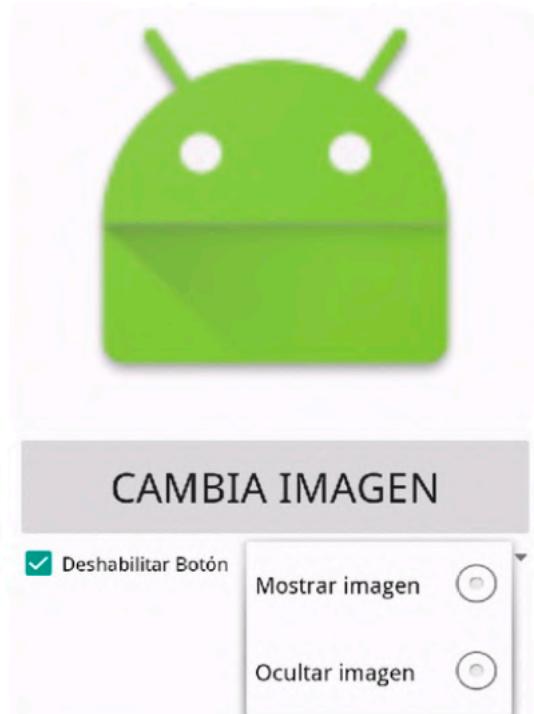


Con esto damos por finalizada este apartado.

#### Tarea UT03.4. Mostrar/Ocultar imágenes.

1. Crea una aplicación con los siguientes elementos:

- Un ImageView con una imagen cargada
- Un botón
- Un checkbox. Al iniciar la aplicación aparece marcado
- Dos radios button. Por defecto aparece marcado "Mostrar" Imagen"



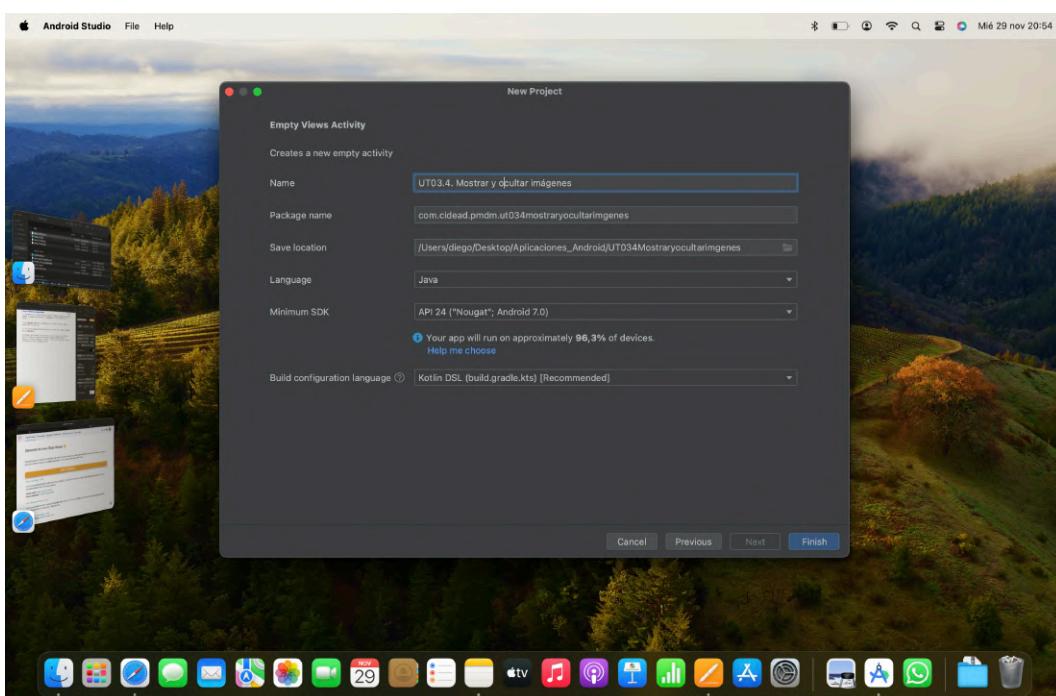
2. Implementa la funcionalidad del botón "Cambiar imagen" para que cambie la imagen mostrada en el ImageView con la captura de su evento.

3. El checkbox habilita o deshabilita el botón "Cambiar imagen" según esté o no activo.

4. Los radios button permiten mostrar u ocultar la imagen cargada en el ImageView.

Lo primero que haremos en este caso, al igual que en los demás, será el crear el proyecto en Android Studio llamado UT03.4.

Mostrar y ocultar imágenes. Elegimos "Java" como lenguaje y como SDK mínimo el API 24 (Android 7,0).



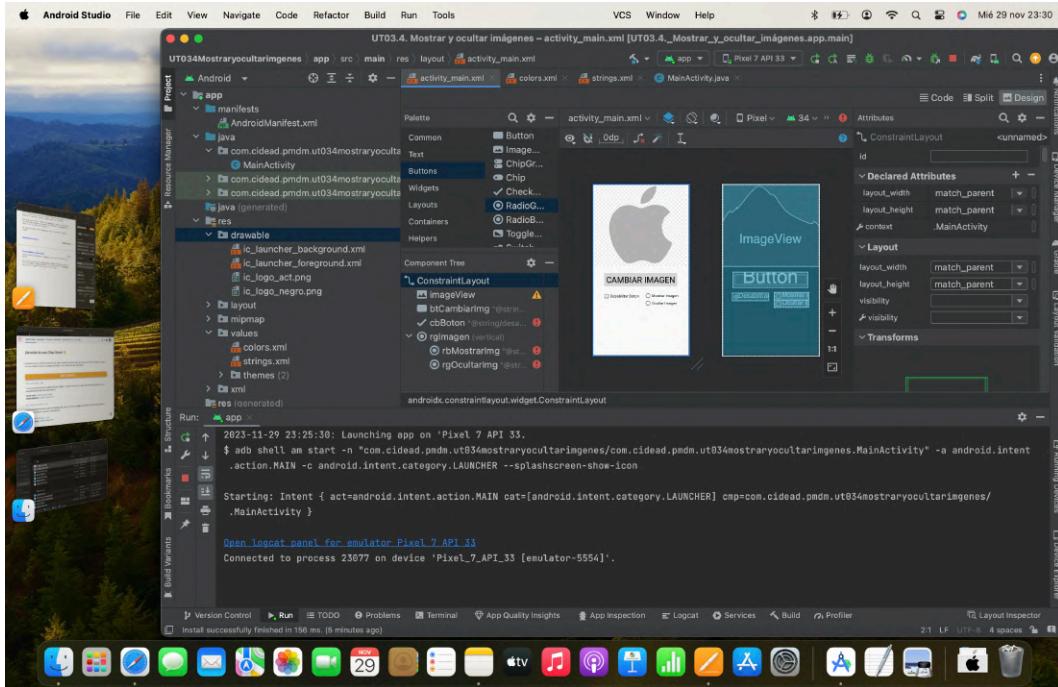
Antes de empezar con la parte gráfica vamos a añadir las imágenes que queremos usar. En nuestro caso hemos elegido dos fotos del logotipo de Apple, uno en gris, que será el que esté cargado por defecto y otro negro, que será el que podamos elegir cuando cliquemos sobre el botón de cambiar imagen. Estas fotos debemos guardarlas en la carpeta drawable, que se encuentra dentro de la carpeta res de nuestro proyecto. Basta con arrastrar a la carpeta en cuestión o copiar y pegar dentro de la misma.

Posteriormente vamos a añadir el ImageView y vamos a seleccionar la imagen que queremos que muestre por defecto. Al ImageView le vamos a dar un tamaño en concreto.

Lo siguiente será crear el botón al cual llamaremos "Cambiar imagen".

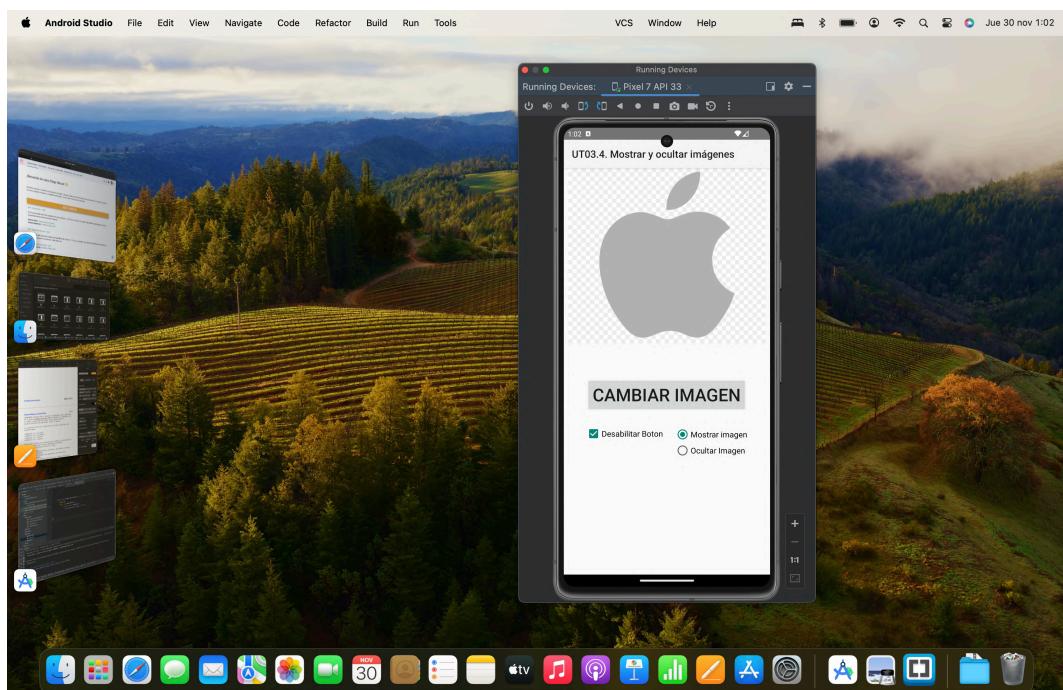
Posteriormente crearemos el checkbox con el texto "Deshabilitar botón".

Por último crearemos dos radiobutton llamados "Mostrar imagen" y "Ocultar imagen" respectivamente y los incluiremos en radiogroup para que solo podamos seleccionar uno a la vez.

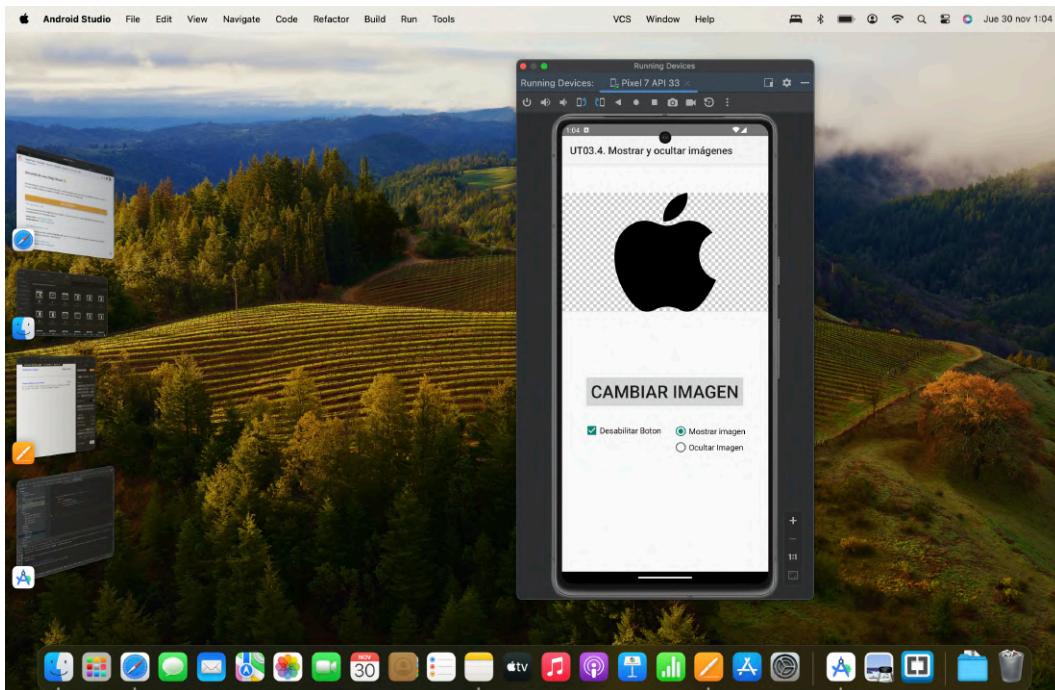


Lo siguiente que haremos será crear el código en el MainActivity.java para añadir las funcionalidades de cada botón.

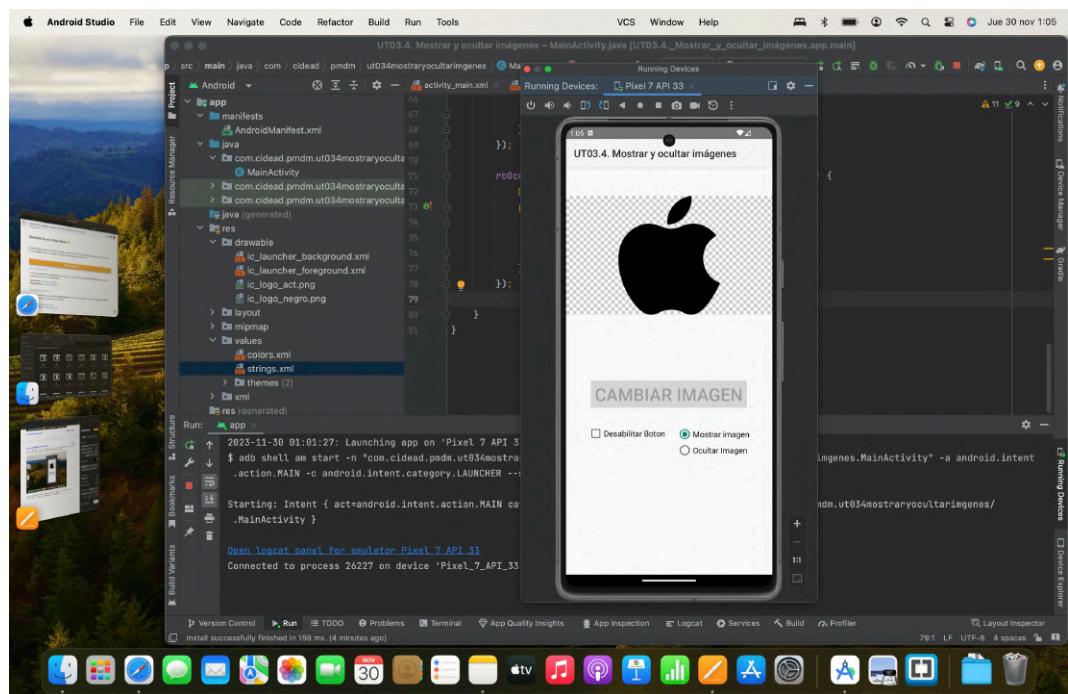
En la siguiente imagen podemos ver como al iniciar la aplicación nos muestra la imagen, el checkbox de activar botón esta marcado y el radiobutton de mostrar imagen aparece seleccionado.



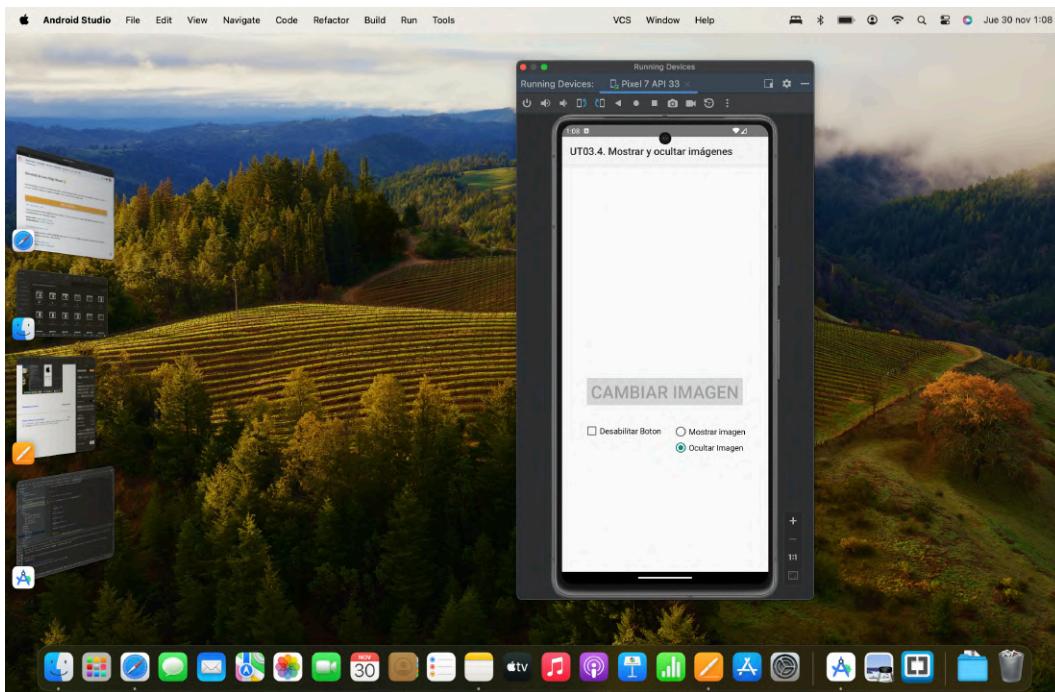
En la siguiente imagen podemos apreciar como ha cambiado la imagen de la app (tras pulsar el botón de cambiar imagen).



En la siguiente imagen podemos apreciar como se deshabilita el botón cambiar imagen al desmarcar el checkbox.



En la siguiente imagen podemos apreciar como desaparece la imagen del textview al marcar el botón de ocultar imagen.



Y con esto damos por finalizado este aparato y por consiguiente la tarea.

**IMPORTANTE:** Entrega todo el proyecto de android studio comprimido. En el archivo MainActivity.java deberá aparecer un comentario con el nombre y apellidos del alumno que ha desarrollado el proyecto. Criterios de puntuación. Total 10 puntos.

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

- EJERCICIO 3.1: 1.5 punto.
- EJERCICIO 3.2: 2.5 puntos.
- EJERCICIO 3.3: 3 puntos.
- EJERCICIO 3.4: 3 puntos.

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.
- Páginas web de estadísticas como gs.statcounter.com.

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; estos te darán todas las claves para acometer la tarea.
- 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 a qué ejercicio pertenecen.
- Debe llevarse a cabo sobre una versión actual de Android.

#### **Indicaciones de entrega.**

Una vez realizada la tarea elaborarás un archivo comprimido con cada proyecto (Tarea UT03.1, Tarea UT03.2, Tarea UT03.3 y Tarea UT03.4).

Muy importante indicar el nombre del alumno en cada `MainActivity.java`, sino no se dará por buena la entrega.

Además, tenéis que hacer un documento de texto indicando los pasos seguidos para programar la app en cada uno de los 4 apartados.

Para su entrega pasarla a pdf con el mismo nombre que se especifica abajo.

Entrega la carpeta comprimida (que contiene los 4 proyectos comprimidos, más el documento pdf explicativo) a través de la plataforma.

El archivo se nombrará siguiendo las siguientes pautas:  
`apellido1_apellido2_nombre_PMDM03_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 tercera unidad de PMDM, debería nombrar esta tarea como...

`sanchez_manas_begona_PMDM03_Tarea`