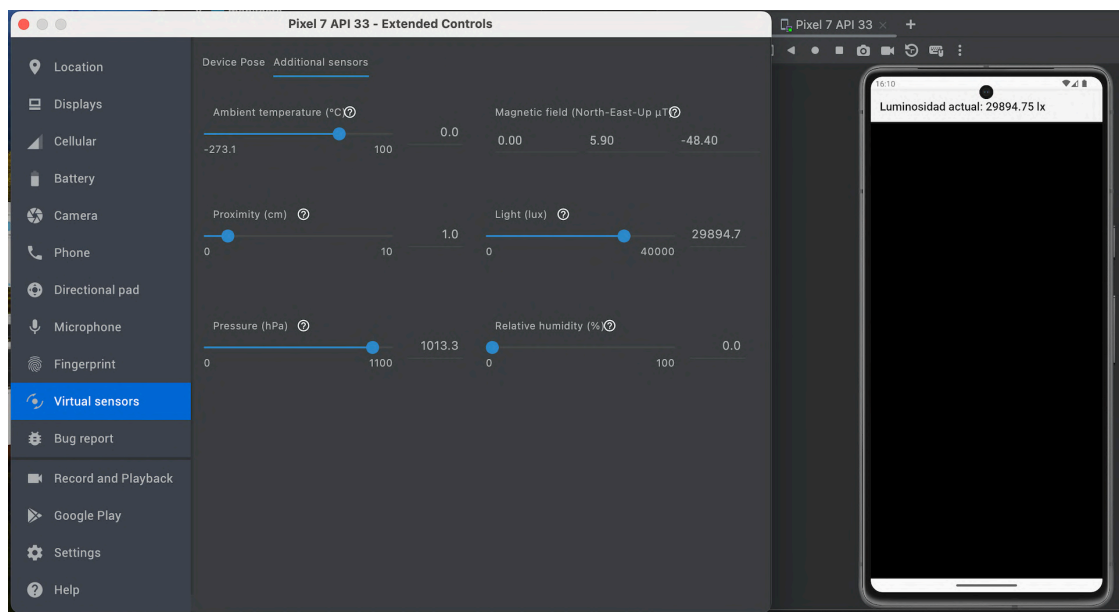
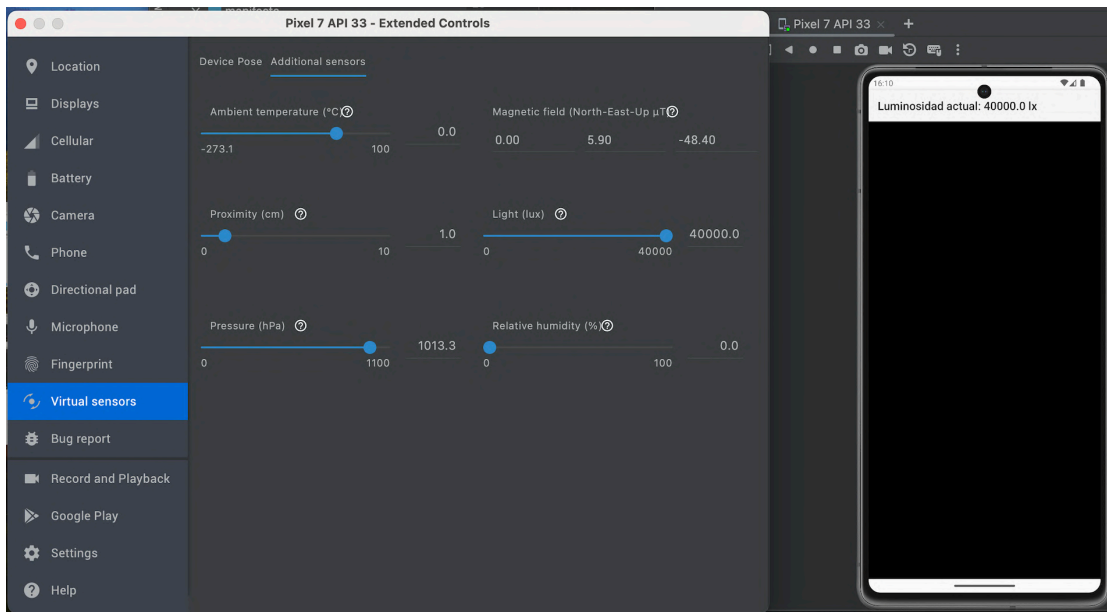


# Tarea 10 para Programación Multimedia y Dispositivos Móviles



Diego Manuel Carrasco Castañares

PMDM10.

- Sensores. Tarea

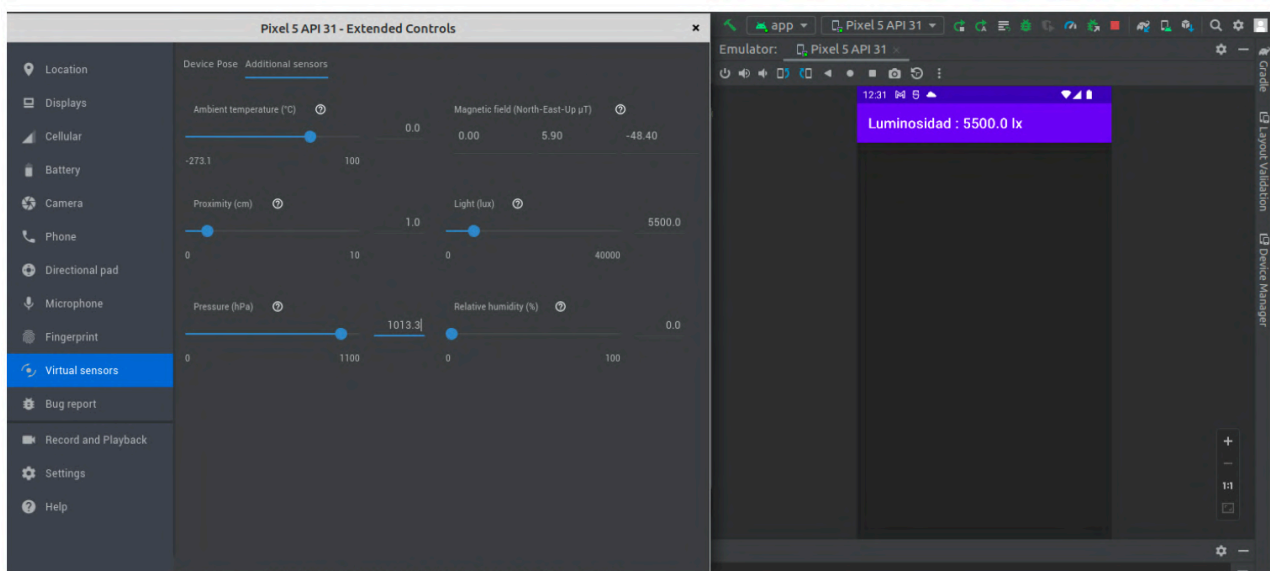
Detalles de la tarea de esta unidad.

Enunciado.

Tarea UT10: Sensor de luminosidad.

Crear una aplicación que maneje un sensor de luminosidad. Dicho sensor es habitual en la mayoría de los teléfonos y tabletas del mercado, por lo que podremos probarlo en nuestro dispositivo real. Otra opción es usar el gestor de sensores de nuestra emulación, al apartado de luminosidad (Light(lux)).

Se pretende que la aplicación muestre en formato texto el valor que recoge el sensor, tal y como muestra la imagen.



Google Developers (Uso educativo nc)

Para mostrar los datos en la parte superior, es necesario utilizar:

```
getSupportActionBar().setTitle("TEXTO");
```

Si decides probar este ejercicio en un dispositivo real, comprobarás que el sensor de luz está en alguna parte próxima a la pantalla, y además, comprobarás de manera significativa cómo cambia el valor de luz registrado cuando acercas o alejas dicho sensor a la fuente de luz.

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 en pantalla el valor de luminosidad captado por el sensor adecuado: 9 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.

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 desees 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 cuando modifiques los valores de luminosidad a través de los sensores virtuales.
- ✓ 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:

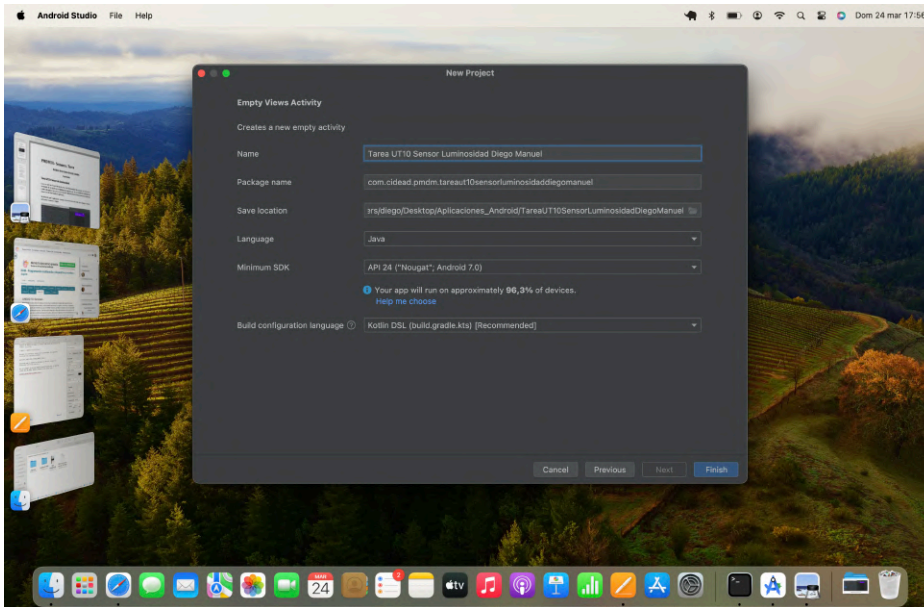
apellido1\_apellido2\_nombre\_PMDM10\_Tarea

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

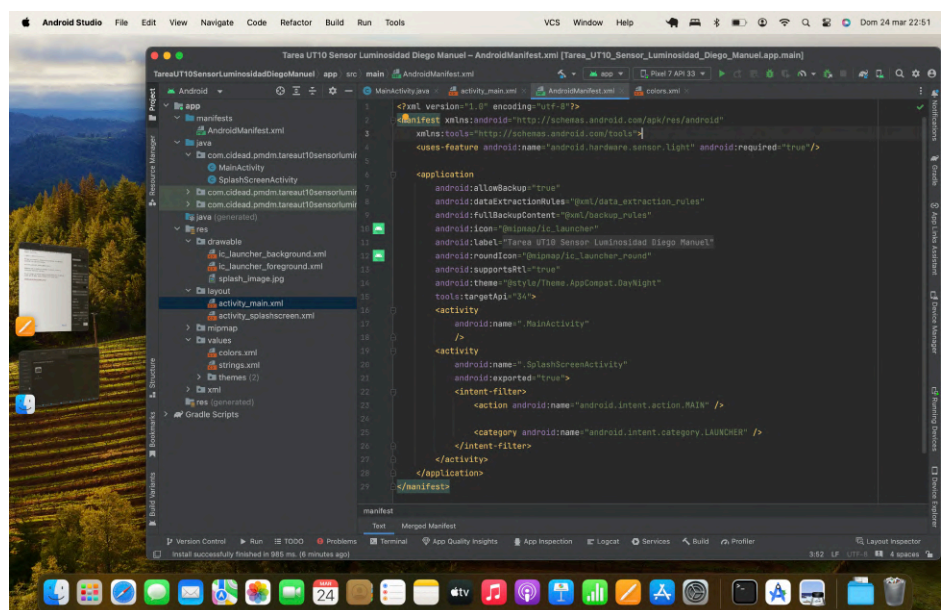
Así por ejemplo la alumna Begoña Sánchez Mañas para la séptima unidad del MP PMDM, debería nombrar esta tarea como...

sanchez\_manas\_begona\_PMDM10\_Tarea

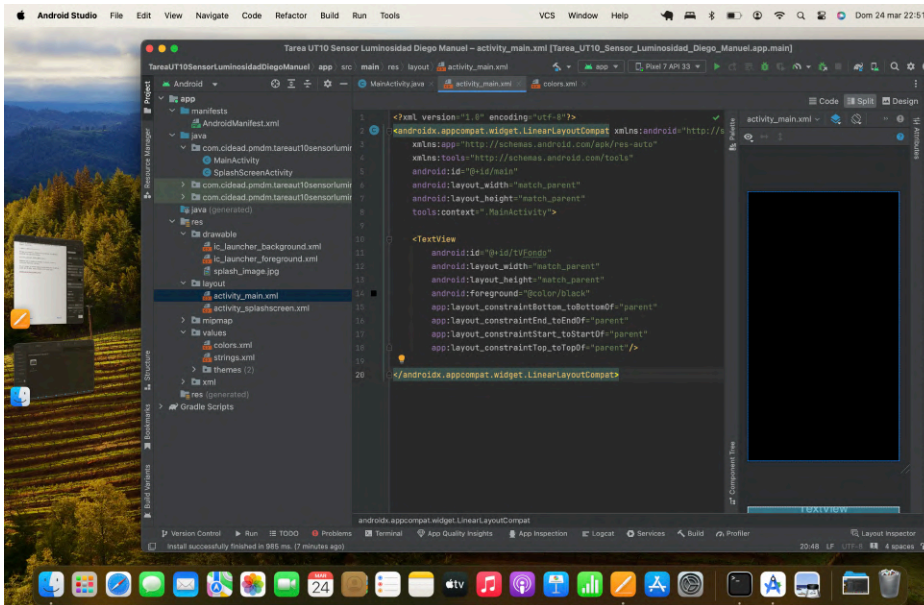
Vamos a empezar creando el proyecto, al cual vamos a llamar *Tarea UT10 Sensor Luminosidad Diego Manuel*.



Empezaremos modificando el `AndroidManifest`, en el cual vamos a indicar que le damos el permiso de acceder al sensor de luminosidad, cambiaremos el tema de la app para que aparezca el título en la parte superior de la misma y, mas adelante, modificaremos para que nos muestre el `SplashScreen` al iniciarla.



En el `activity_main` vamos a crear solamente un `TextView` para darle fondo a la app, el cual será negro, tal como se ve en la imagen de la tarea.



En MainActivity vamos a declarar tres atributos, uno de tipo sensor, otro de tipo `SensorEventListener` y otro de tipo `SensorManager`, los cuales serán privados.

Tras ello vamos a componer el código que contiene el método `onCreate`.

En el vamos a obtener el servicio del `SensorManager`, el cual le pasaremos al `Sensor` para instanciarlo.

Tras ello comprobaremos, mediante un `if`, si el sensor esta instanciado a `null`, en cuyo caso mostrará un `Toast` en el que indicará que no se ha encontrado el sensor y finalizara la actividad.

En caso de no estar instanciado a `null`, quiere decir que ha puede acceder al sensor de luminosidad por lo que instanciaremos un nuevo `SensorEventListener`.

En dicho evento sobreescribiremos el método `onSensorChanged` para capturar cuando el sensor varía.

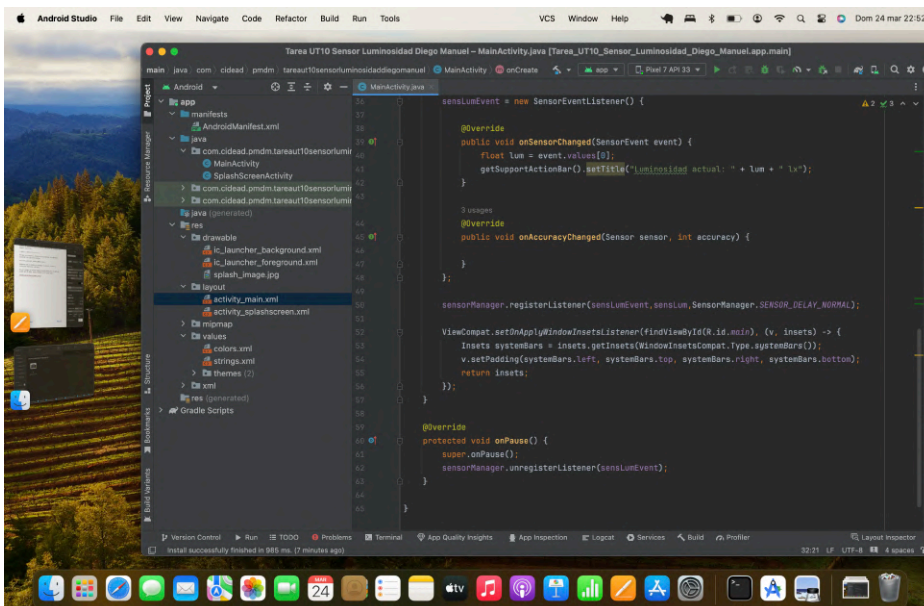
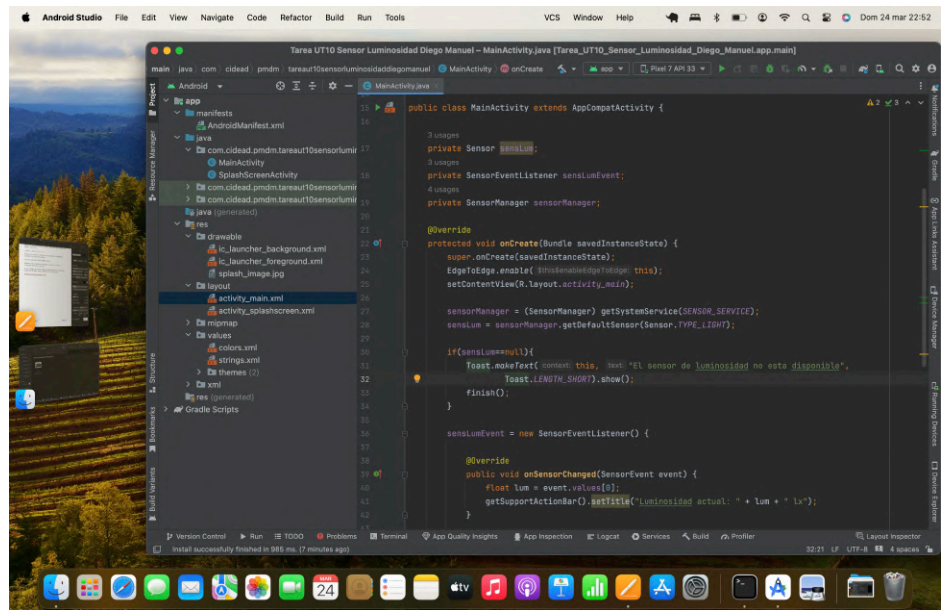
Para ello declararemos dentro de él un `float` que capturará el valor del sensor y utilizaremos el método indicado en la tarea (`getSupportActionBar().setTitle()`; para mostrar en la la `ActionBar` la luminosidad captada por el mismo.

El método `onAccuracyChanged` lo dejaremos tal cual está.

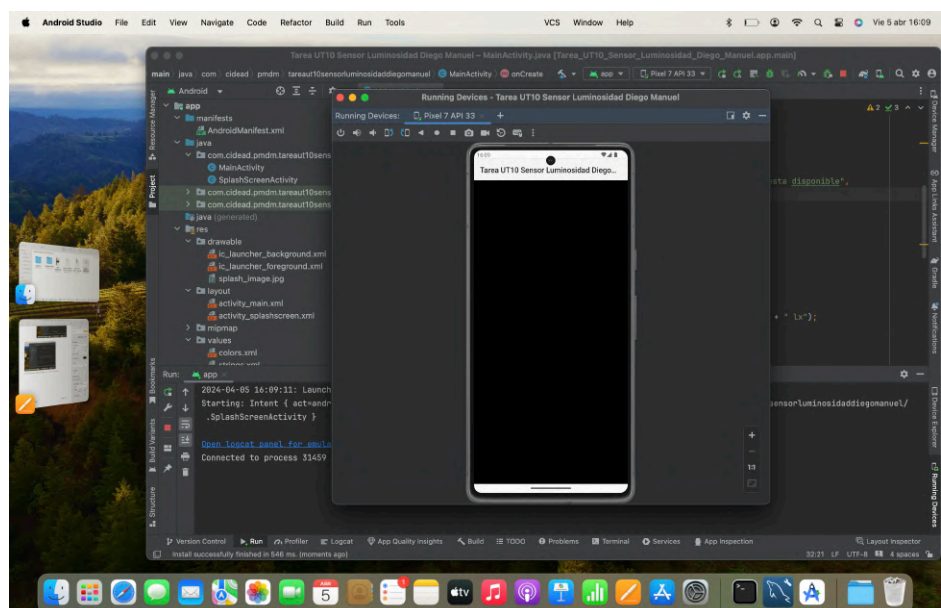
También registraremos el listener para capturar el evento del sensor de luminosidad, a la cual, mediante la constante `SENSOR_DELAY_NORMAL` le daremos una tasa de actualización normal.

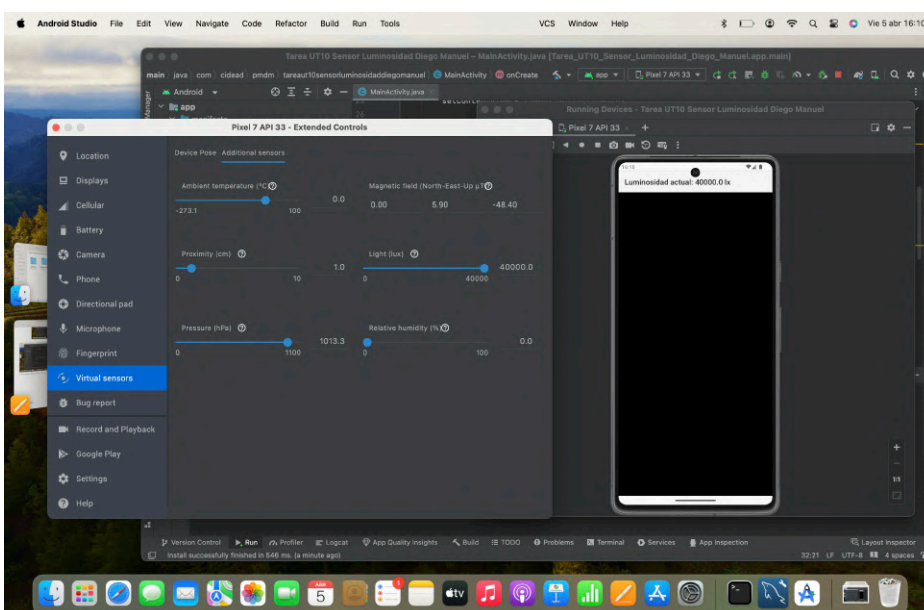
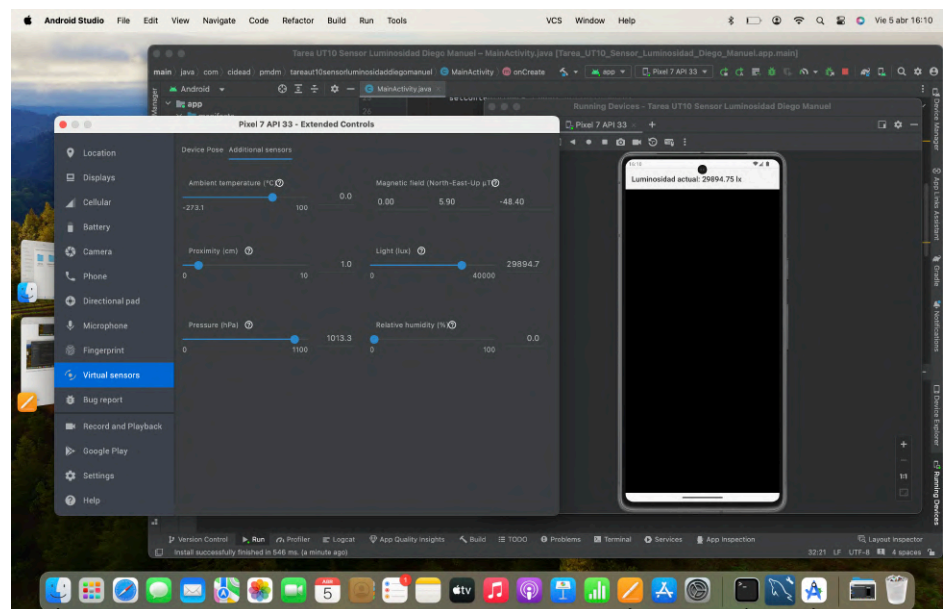
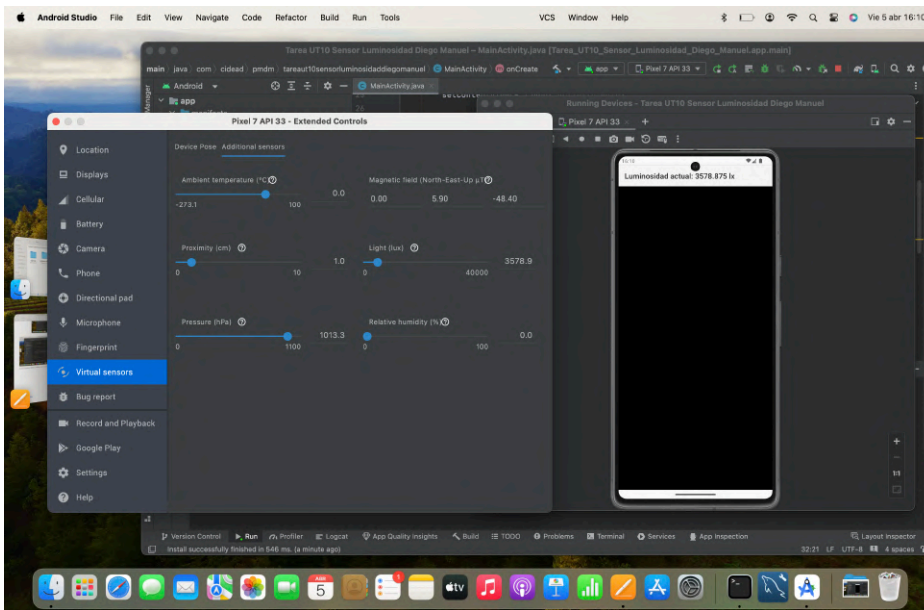
Por último sobreescribiremos el método `onPause`, el cual detendrá el evento de escucha del sensor una vez terminada la actividad.





En las siguientes imágenes podemos ver la app funcionando.







Adjunto un video donde también se puede apreciar el funcionamiento de la app.

Y con esto damos por finalizada la tarea.