**Requerimiento Bloque 1:**

Generá un archivo .doc con lo planteado en la consigna. Adjuntá los resultados de este ejercicio en el proyecto final que entregarás al final de este recorrido.

**Ejercicio práctico**

**Introducción a Android**

Crear una aplicación utilizando Android Studio con las siguientes especificaciones:

* Nombre: AppPractica1
* Nombre del paquete: com.curso.android.app.practica.uno
* Versión mínima a soportar: Android 5
* Basa en un template del tipo: Vista Primaria con Detalle

***Actividad 1***

* Crear un emulador del tipo Tablet, con un tamaño de pantalla de al menos 8”.
* Correr la aplicación en el emulador en orientación Portrait (vertical).
* Correr la aplicación en el emulador en orientación Landscape (horizontal).

***Actividad 2***

* Crear un emulador del tipo Teléfono.
* Correr la aplicación en el emulador en orientación Portrait (vertical).
* Correr la aplicación en el emulador en orientación Landscape (horizontal).

***Actividad 3***

En ambos los casos:

* Indicar que configuración tiene cada emulador creado (versión de Android, tamaño de pantalla, memoria RAM).
* Realizar capturas de pantalla de cómo se ven todas las pantallas de la aplicación en cada configuración.

***Actividad 4***

* ¿Qué diferencia observás en el funcionamiento de la aplicación en una tablet con orientación Landscape y un teléfono con orientación Portrait?

***Actividad extra***

* ¿En qué archivo está el nombre de tu aplicación? Probá cambiarlo y correr la aplicación nuevamente.
* ¿Y el ícono?

**Creando App**

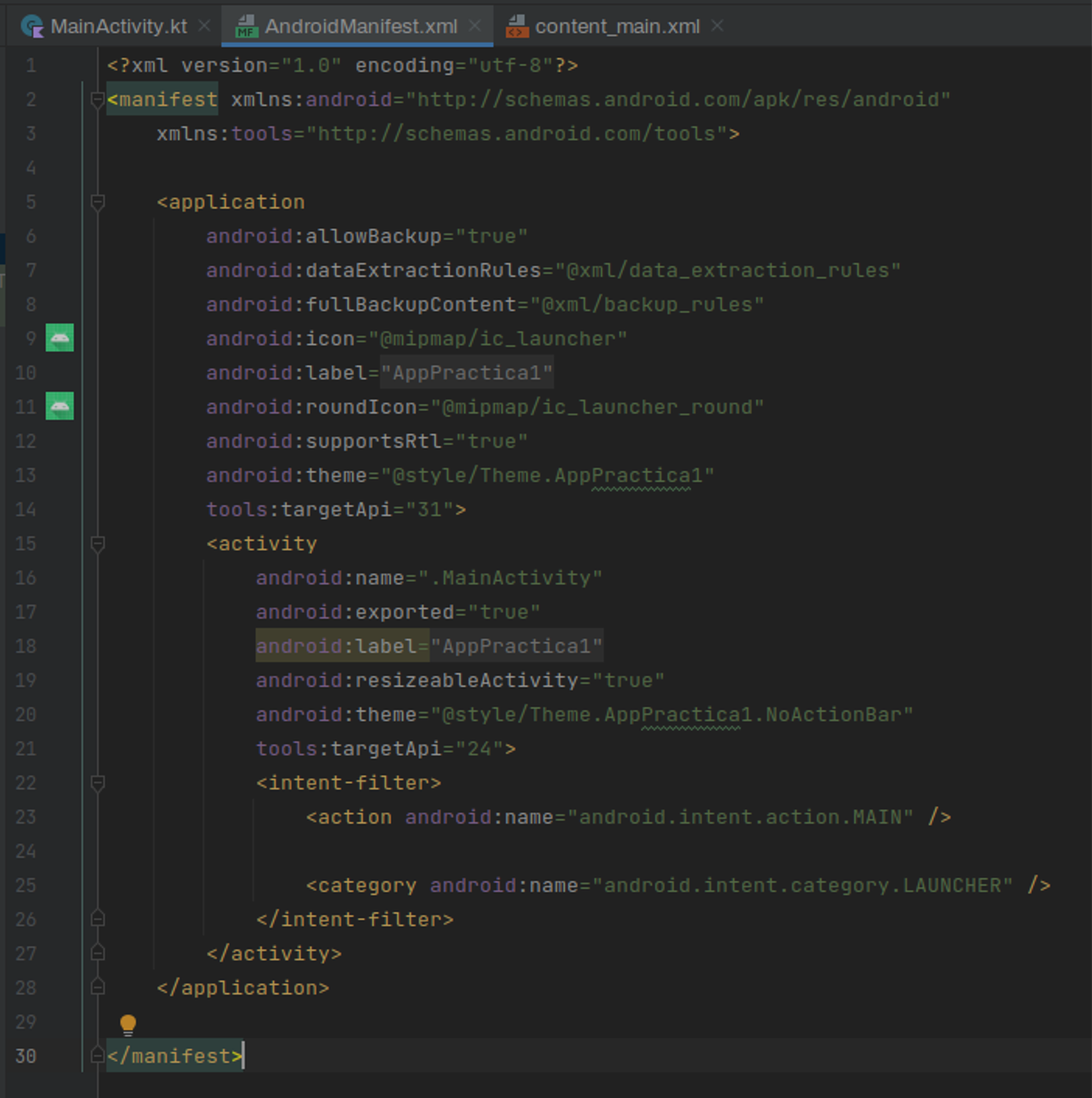
Nombre: AppPractica1

Nombre del paquete: com.curso.android.app.practica.uno

Versión mínima a soportar: Android 5

Basada en un template del tipo: Vista Primaria con Detalle

Abrimos Android Studio, vamos a New Project, y buscamos la plantilla Vista Primaria con Detalle (Primary/Detail Flow). Se nos abre un cuadro donde llenamos en los campos con el nombre de la App (AppPractica1) y los demás requerimientos especificados arriba. Apretamos *Finish* y se va a generar el proyecto, mientras muestra *Loading* lo va a indexar, compilar y luego podremos correr la aplicación.



**Actividad 1 = Crear Emulador tipo Tablet**

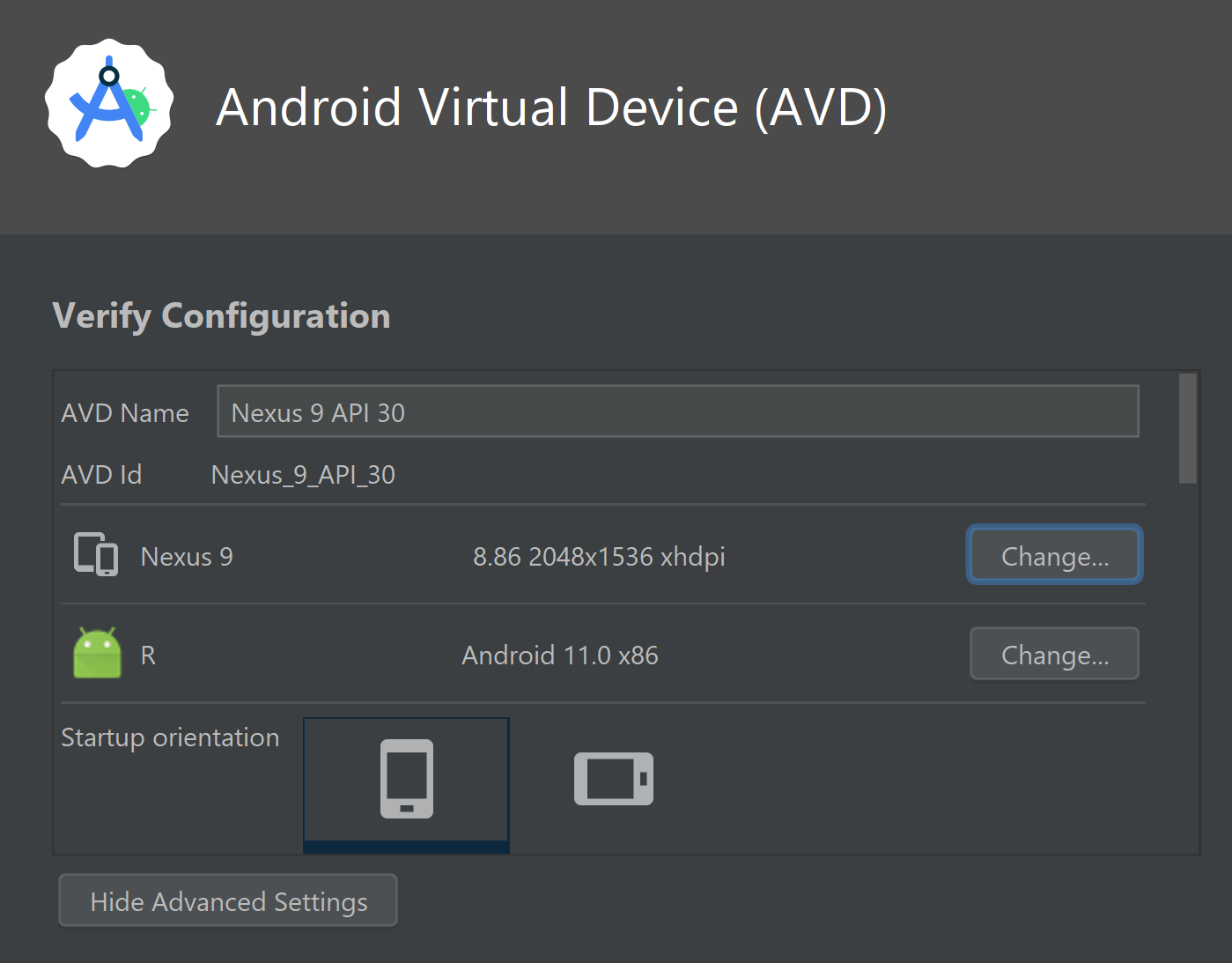
**con Actividad 3; Indicar configuración (versión de Android, tamaño de pantalla, memoria RAM). Realizar capturas de pantalla.**

**AVD Name:** Nexus 9API 30

**Tamaño de pantalla:** 8.86” (2048x1536 xhdpi)

**Version de Android:** Android 11 R x86

**RAM:** 2048 MB



Properties

avd.ini.displayname Nexus 9 API 30

avd.ini.encoding UTF-8

AvdId Nexus\_9\_API\_30

disk.dataPartition.size 32G

fastboot.chosenSnapshotFile

fastboot.forceChosenSnapshotBoot no

fastboot.forceColdBoot no

fastboot.forceFastBoot yes

hw.accelerometer yes

hw.arc false

hw.audioInput yes

hw.battery yes

hw.camera.back virtualscene

hw.camera.front emulated

hw.cpu.ncore 2

hw.device.hash2 MD5:1d017f4d8ab4265c9a5118f248f36f04

hw.device.manufacturer Google

hw.device.name Nexus 9

hw.dPad no

hw.gps yes

hw.gpu.enabled yes

hw.gpu.mode auto

hw.initialOrientation Portrait

hw.keyboard yes

hw.lcd.density 320

hw.lcd.height 1536

hw.lcd.width 2048

hw.mainKeys no

hw.ramSize 2048

hw.sdCard yes

hw.sensors.orientation yes

hw.sensors.proximity no

hw.trackBall no

image.androidVersion.api 30

image.sysdir.1 system-images\android-30\google\_apis\x86\

PlayStore.enabled false

runtime.network.latency none

runtime.network.speed full

showDeviceFrame yes

skin.dynamic yes

tag.display Google APIs

tag.id google\_apis

vm.heapSize 192

**Correr la aplicación en el emulador en orientación Portrait (vertical)**

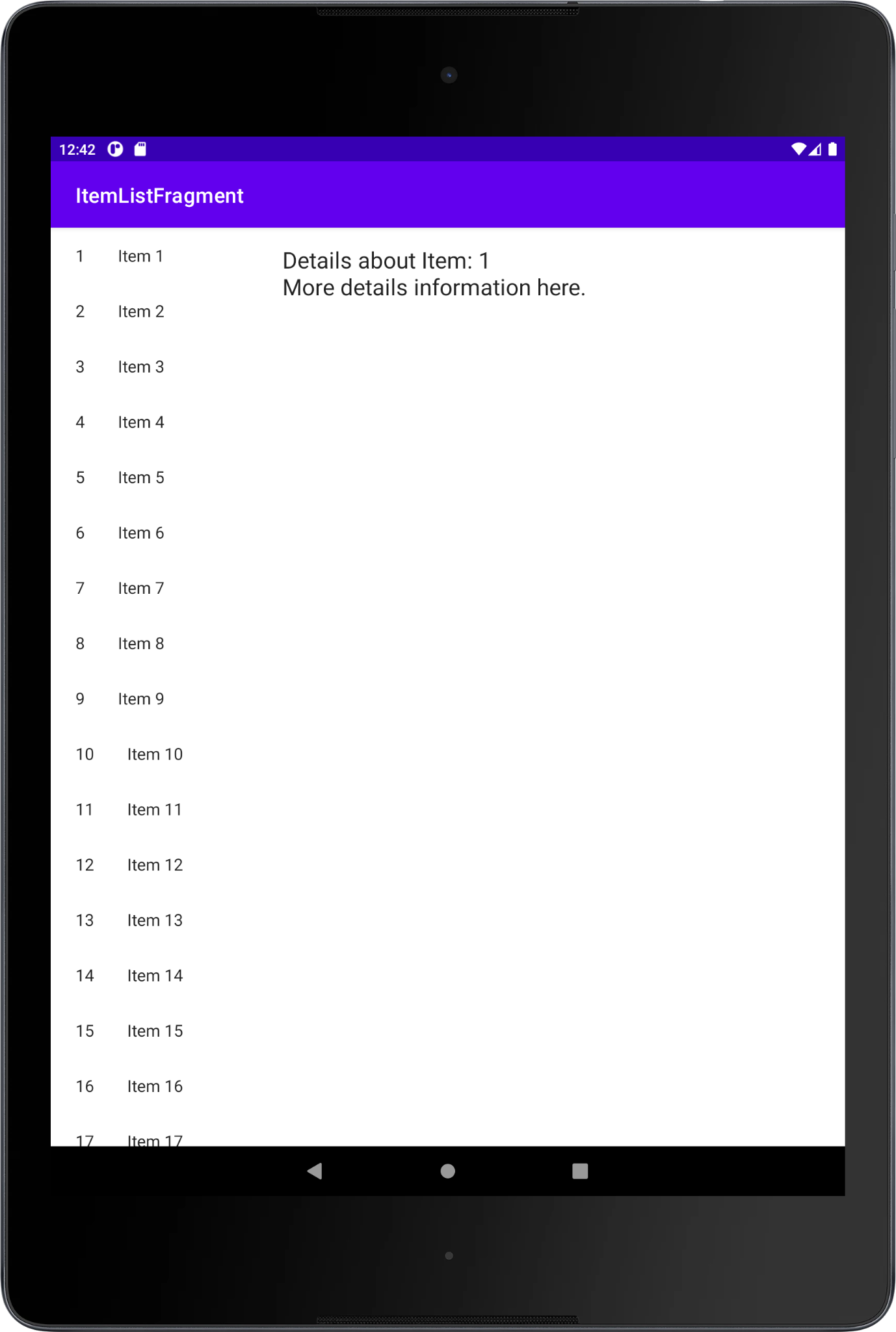
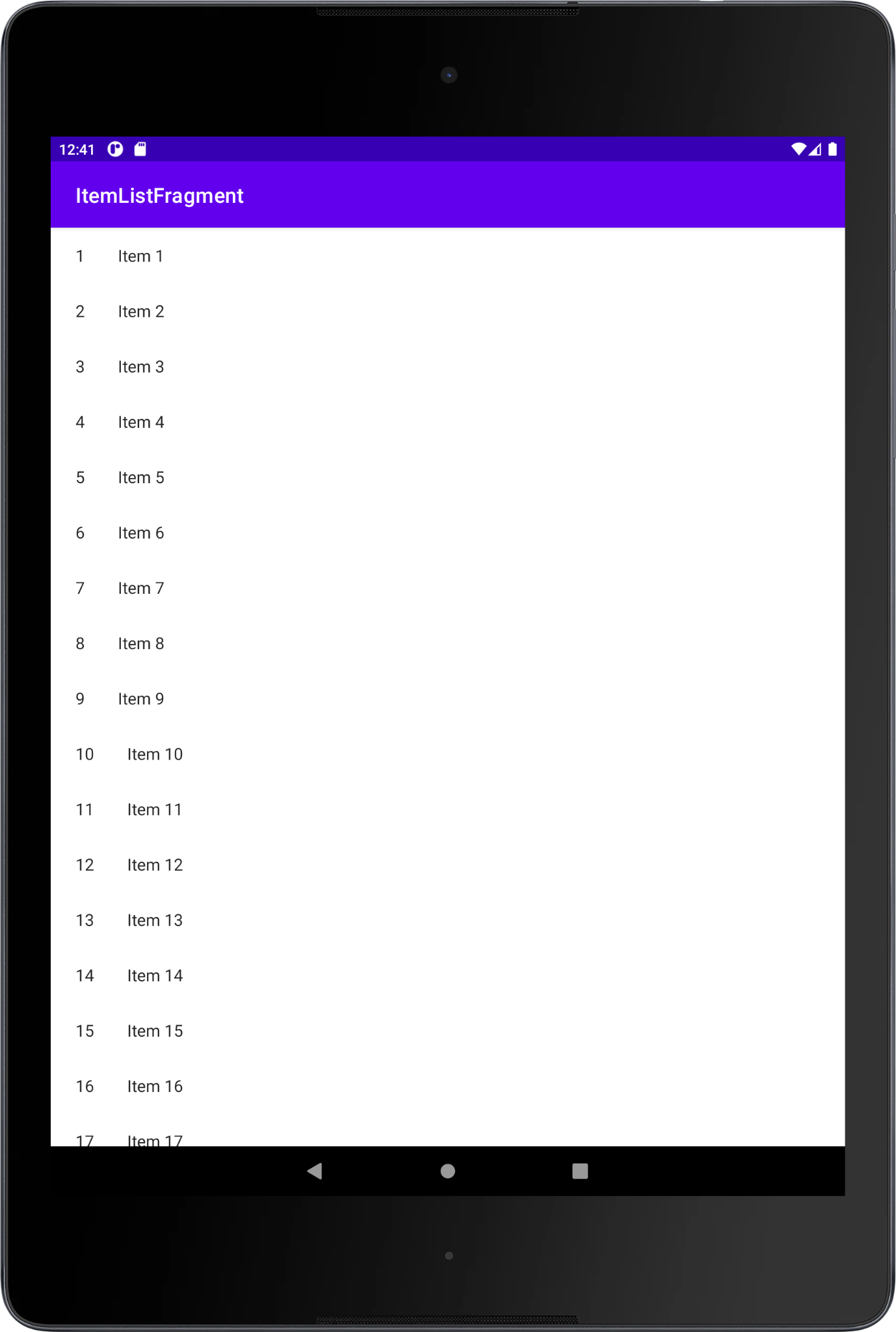
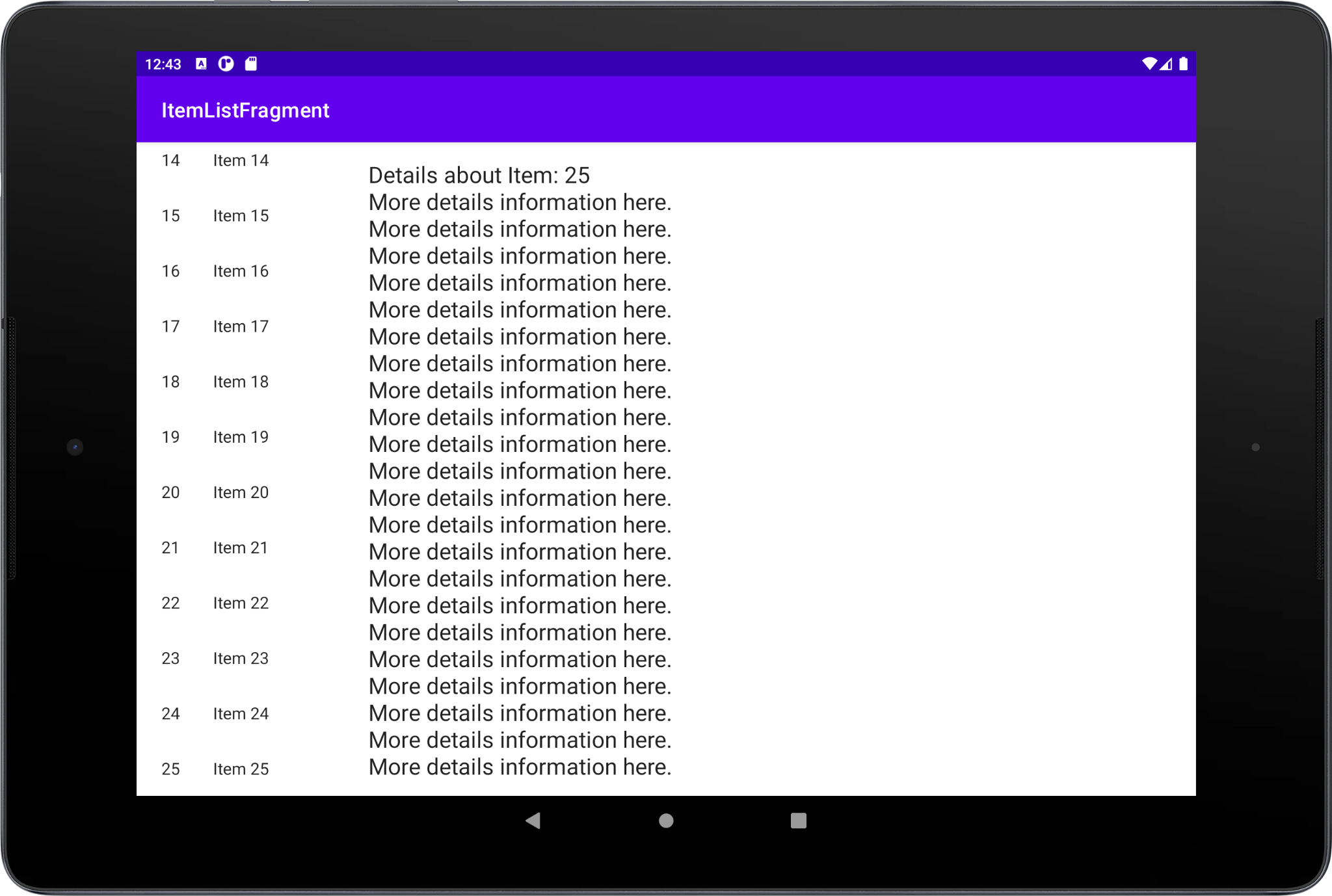
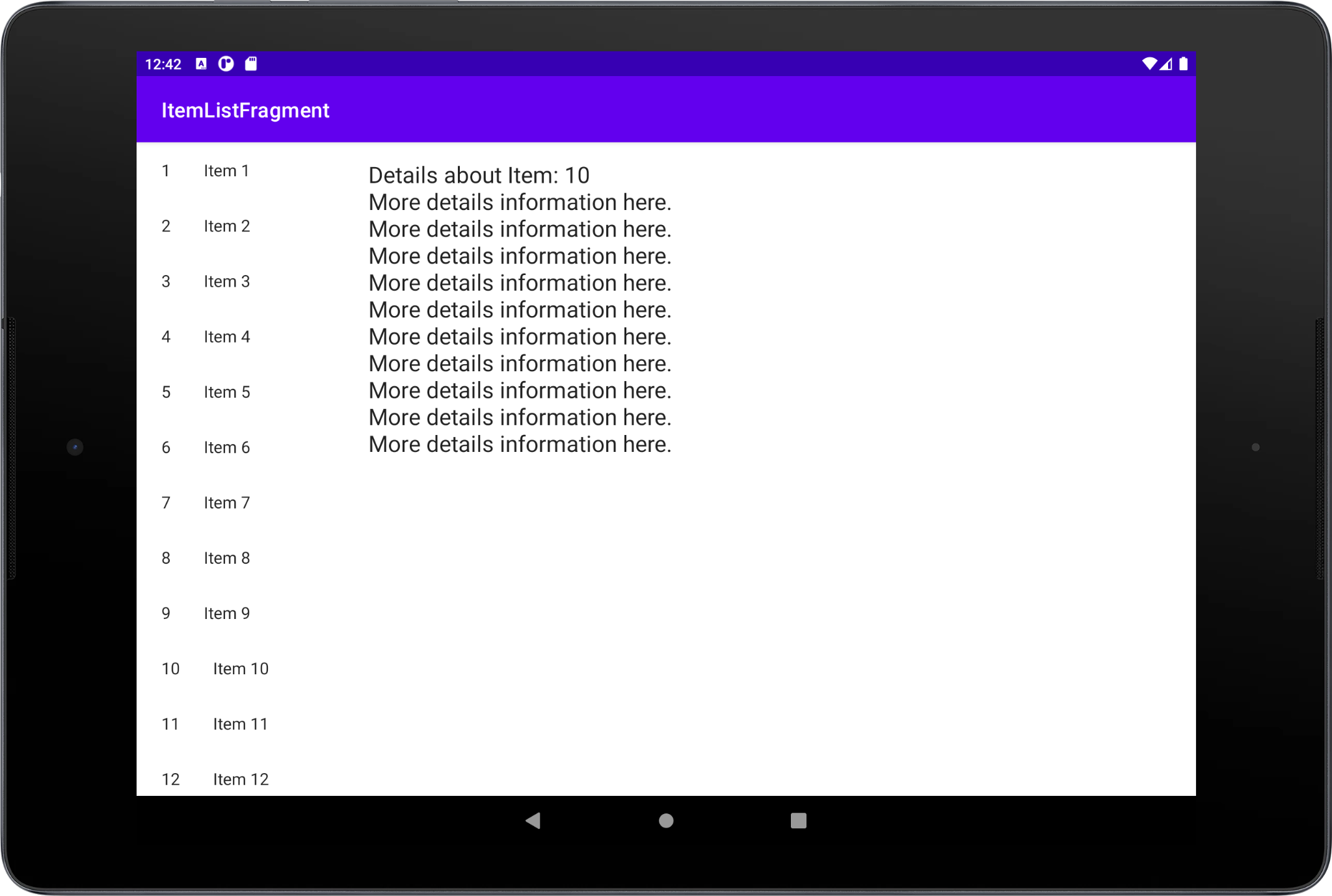


Imagen izquierda: Lista de Items siempre visible, con espacio vacío al lado.

Imagen derecha: Lista de Items y espacio de Detalles en la misma pantalla, dividida.

**Correr la aplicación en el emulador en orientación Landscape (horizontal)**



En las dos imágenes se muestra que al transicionar en la rotación a horizontal hay aún más espacio, de sobra, para mostrar los dos fragments, Lista y Detalles, al mismo tiempo.

**Actividad 2 = Crear Emulador tipo Teléfono Móvil**

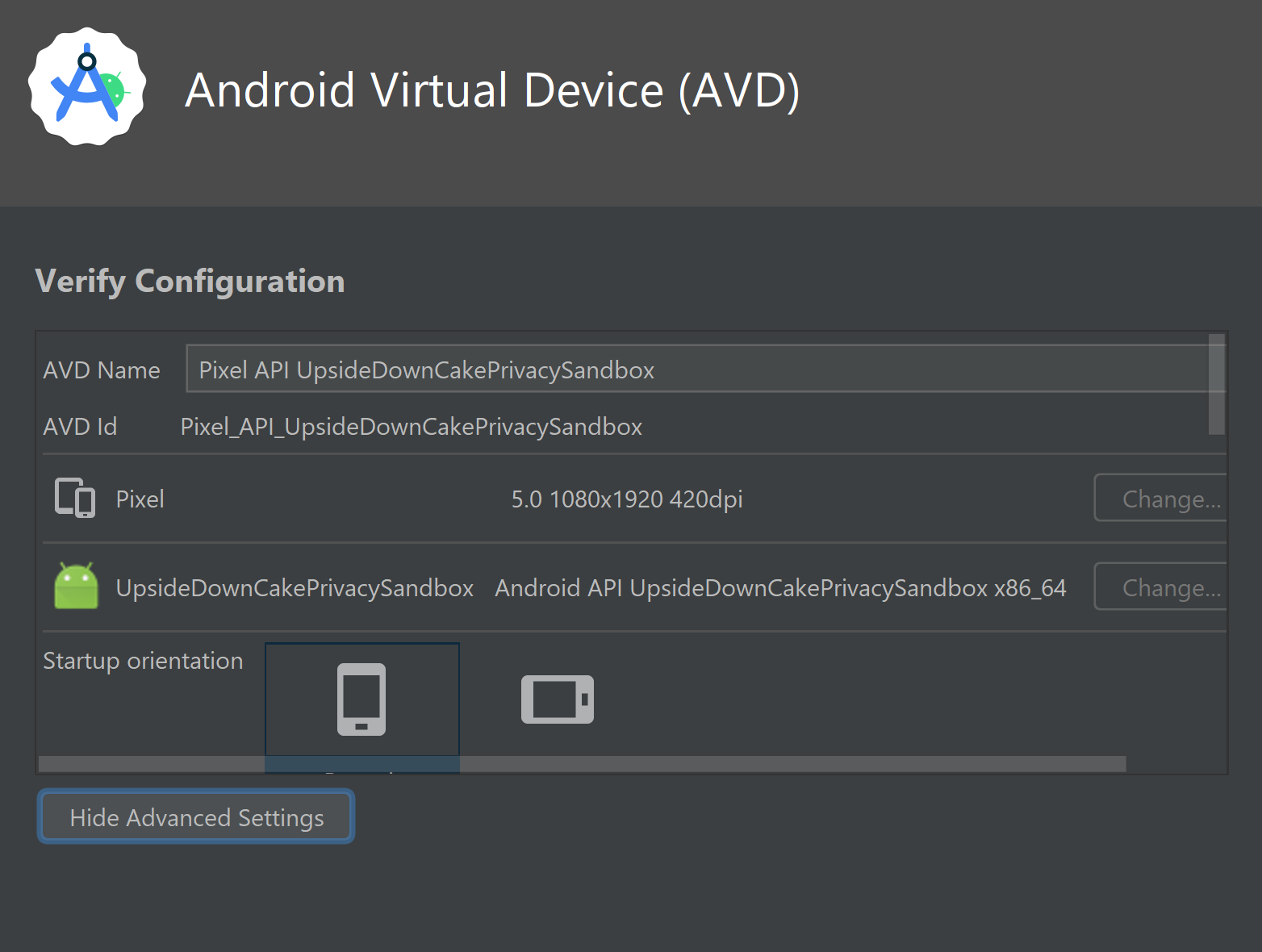
**con Actividad 3; Indicar configuración (Android, tamaño de pantalla, memoria RAM). Realizar capturas de pantalla.**

**AVD Name:** Google PixelAPI 33

**Tamaño de pantalla:** 5.0” (2048x1536 xhdpi)

**Version de Android:** Android 14 Upside Down Cake Privacy Sandbox

**RAM:** 1536 MB



Properties

avd.ini.displayname Pixel API UpsideDownCakePrivacySandbox

avd.ini.encoding UTF-8

AvdId Pixel\_API\_UpsideDownCakePrivacySandbox

disk.dataPartition.size 6442450944

fastboot.chosenSnapshotFile

fastboot.forceChosenSnapshotBoot no

fastboot.forceColdBoot no

fastboot.forceFastBoot yes

hw.accelerometer yes

hw.arc false

hw.audioInput yes

hw.battery yes

hw.camera.back virtualscene

hw.camera.front emulated

hw.cpu.ncore 2

hw.device.hash2 MD5:55acbc835978f326788ed66a5cd4c9a7

hw.device.manufacturer Google

hw.device.name pixel

hw.dPad no

hw.gps yes

hw.gpu.enabled yes

hw.gpu.mode auto

hw.initialOrientation Portrait

hw.keyboard yes

hw.lcd.density 420

hw.lcd.height 1920

hw.lcd.width 1080

hw.mainKeys no

hw.ramSize 1536

hw.sdCard yes

hw.sensors.orientation yes

hw.sensors.proximity yes

hw.trackBall no

image.androidVersion.api 33

image.androidVersion.codename UpsideDownCakePrivacySandbox

image.sysdir.1 system-images\android-UpsideDownCakePrivacySandbox\google\_apis\_playstore\x86\_64\

PlayStore.enabled true

runtime.network.latency none

runtime.network.speed full

showDeviceFrame yes

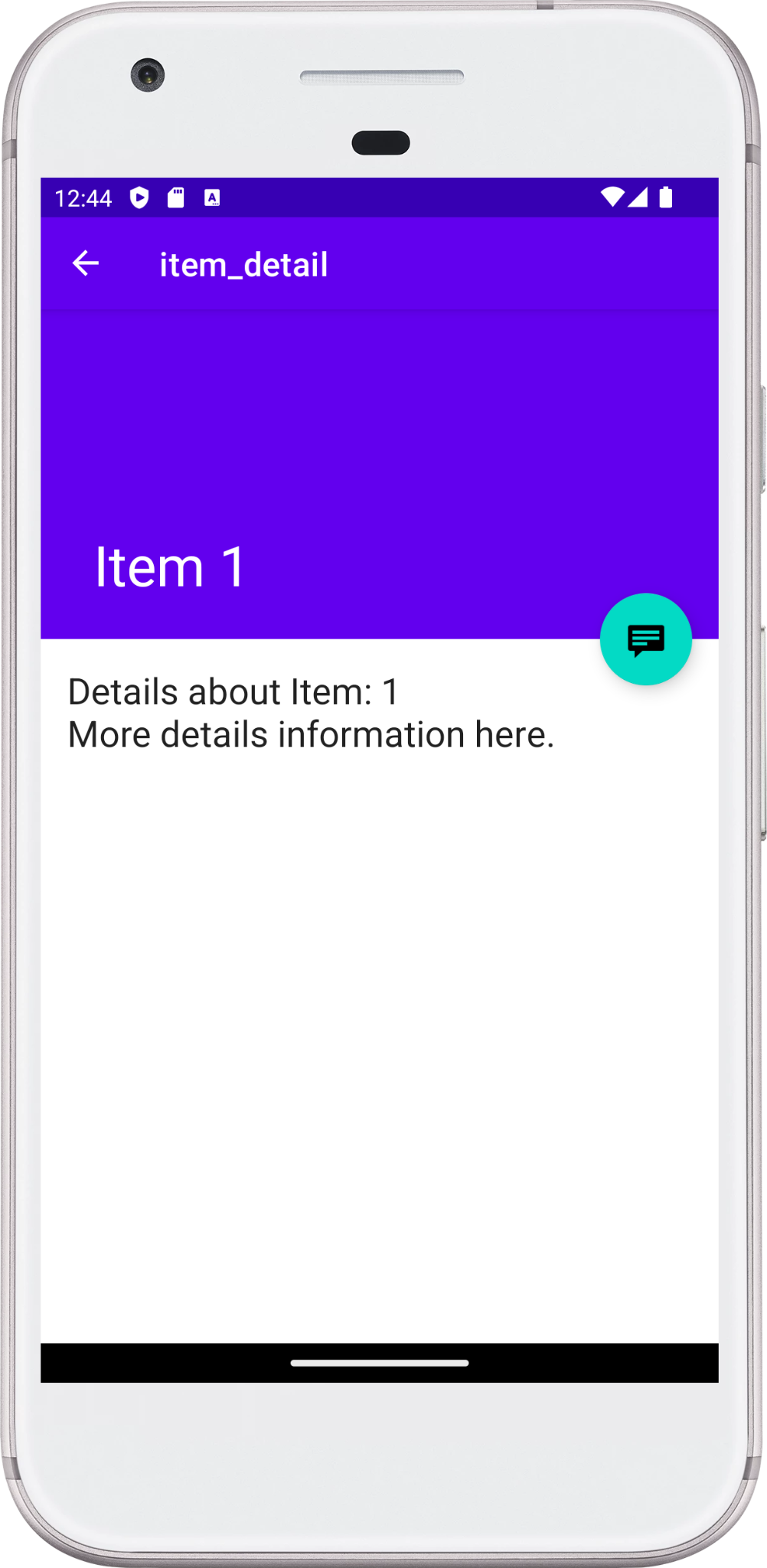
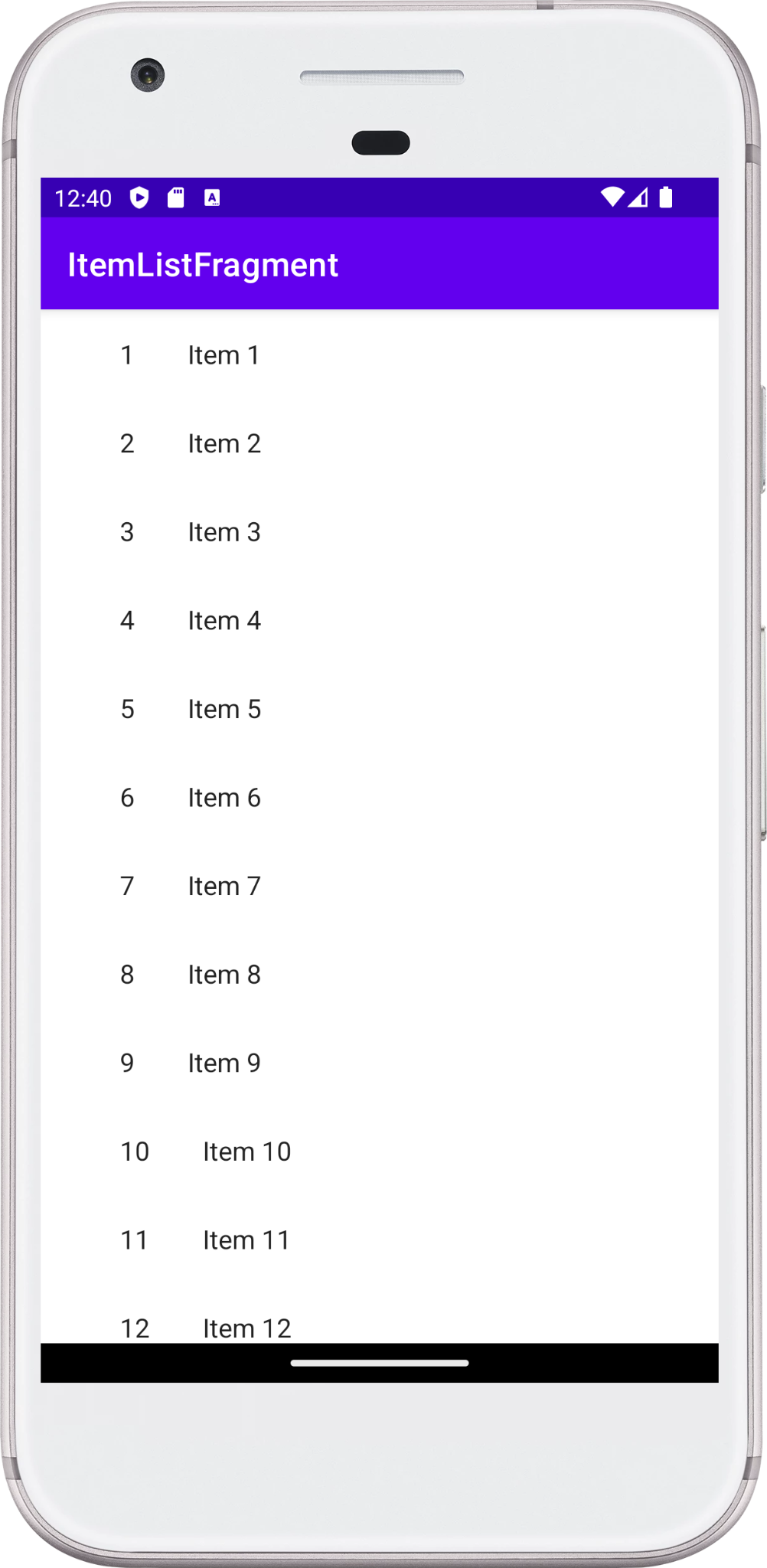
skin.dynamic yes

tag.display Google Play

tag.id google\_apis\_playstore

vm.heapSize 228

**Correr la aplicación en el emulador en orientación Portrait (vertical)**

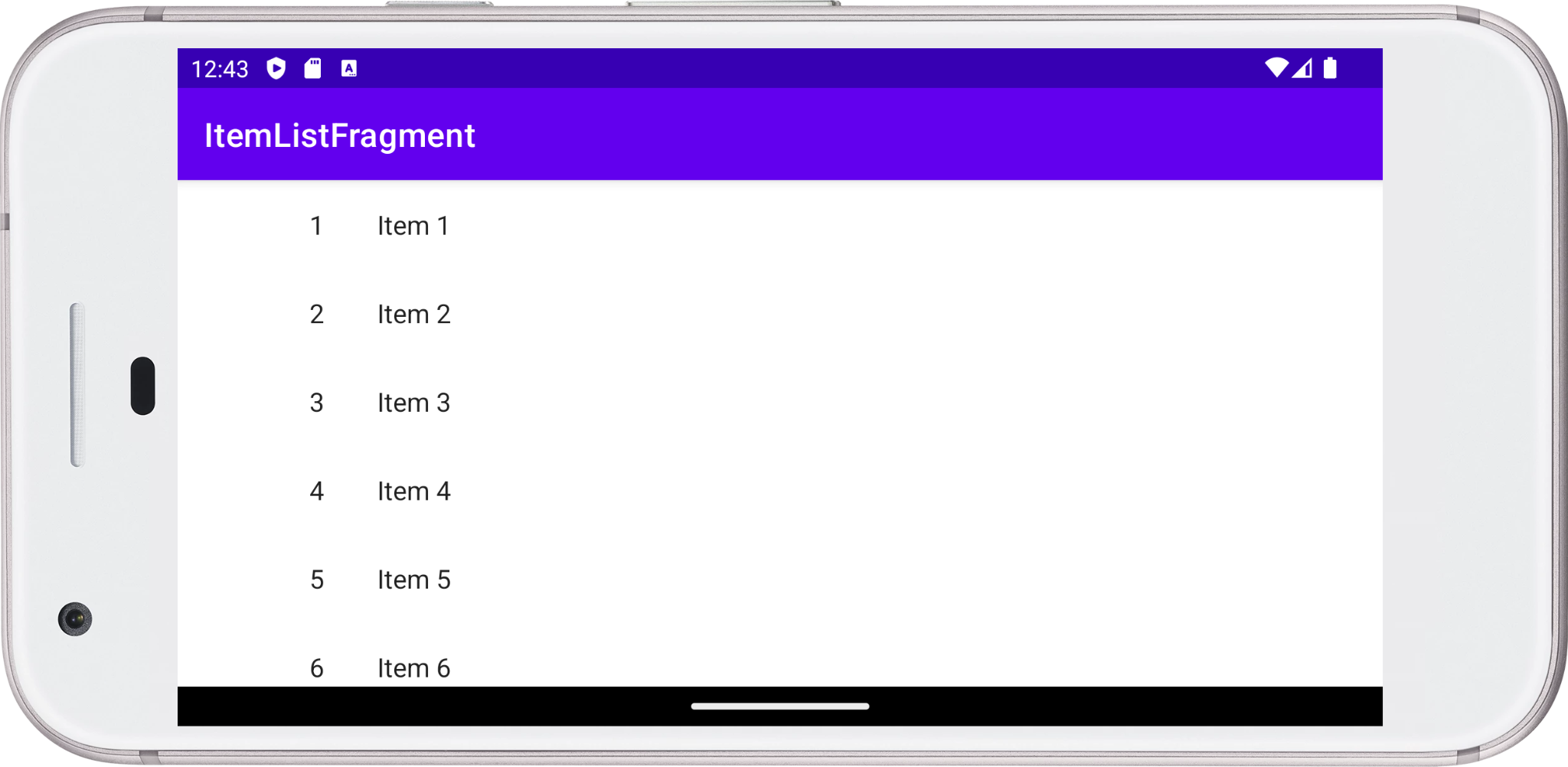
****

En la imagen izquierda se puede ver el fragment de la Lista de Items (ItemListFragment), con alineación izquierda.

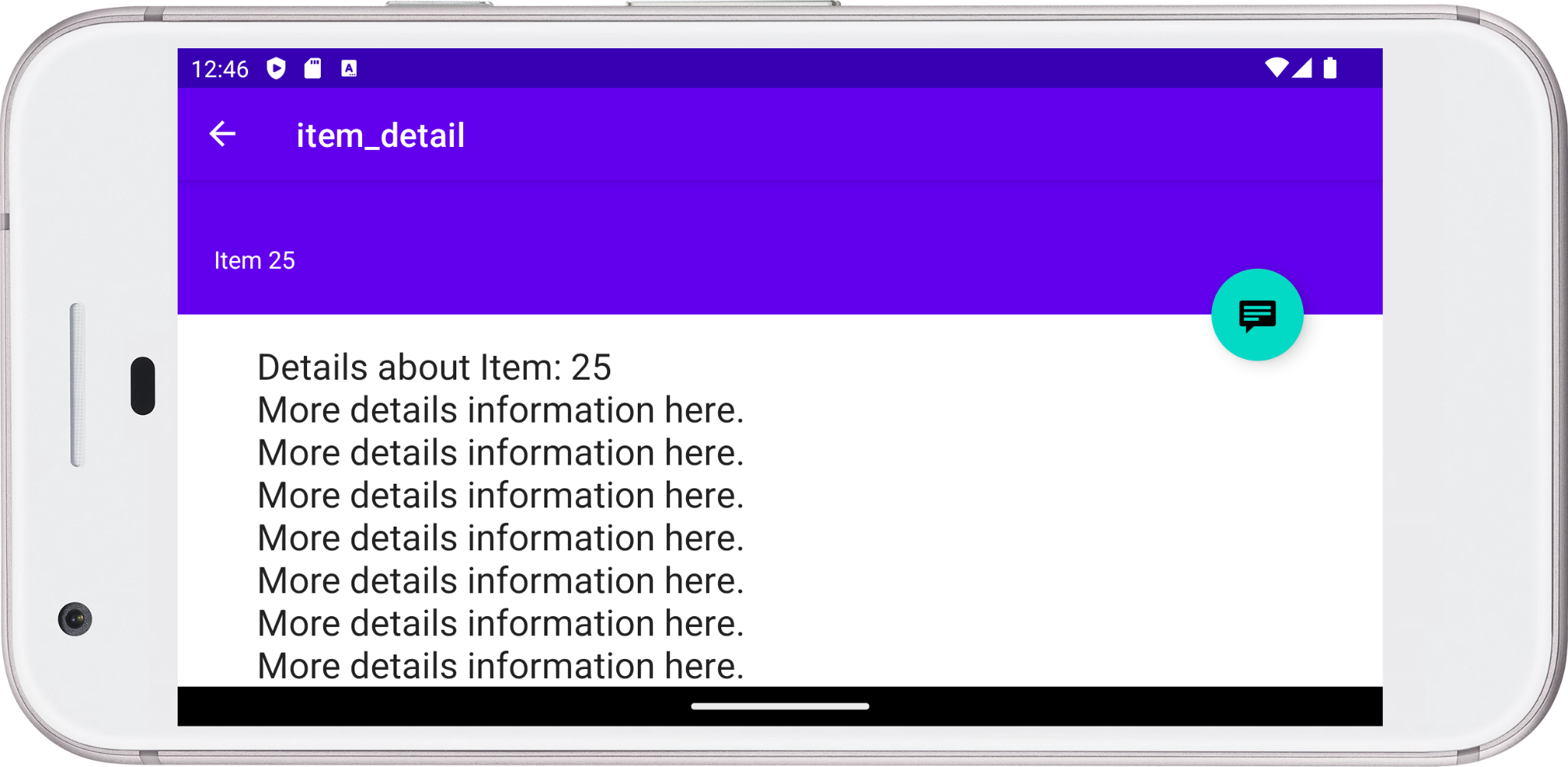
En la imagen derecha se aprecia que al cliquear el Item 1 se despliega una pantalla nueva para mostrar los Detalles (item\_detail).

En Orientación Portrait del móvil hay poco espacio para la Lista y los Detalles, así que presenta dos vistas distintas para la misma actividad y los mismos fragments que en la Tablet.

**Correr la aplicación en el emulador en orientación Landscape (horizontal)**

****

En la imagen, igual que la orientación vertical, se ve la Lista de Items (ItemListFragment) en alineación izquierda, con suficiente espacio al lado.

****

Finalmente, en este último horizontal, se aprecia que al cliquear un item cualquiera (el 25) se despliega una pantalla nueva para el fragment de los Detalles (item\_detail).

Igual que en la orientación vertical, aquí también se muestran Lista y Detalles en dos vistas distintas.

**Actividad 4: Responder**

**¿Qué diferencia observás en el funcionamiento de la aplicación en una tablet con orientación Landscape y un teléfono con orientación Portrait?**

Hay diferencias notorias. Por ejemplo, en la transición a horizontal (Landscape) de la tablet hay mucho espacio para mostrar las fragments de Lista de Opciones y Detalles, a la vez. Y como están en alineación a izquierda, queda todavía espacio en blanco a la derecha de la pantalla.

El orden visual de UI es en GridLayout de 2 columnas, en las que una (Lista) se muestra siempre, y la otra (Detalles) varía de acuerdo a los items seleccionados.

En cambio que en elteléfono con orientación vertical (Portrait) se puede ver el fragment de la Lista de Items (ItemListFragment) fijo. Al seleccionar un ítem aparece una pantalla nueva para mostrar los Detalles (item\_detail).

En esta orientación hay poco espacio para un Layout de dos columnas (la Lista y los Detalles), así que presenta dos vistas distintas para la misma actividad y los mismos fragments que en la Tablet.

**Actividad Extra**

**¿En qué archivo está el nombre de tu aplicación? Probá cambiarlo y correr la aplicación nuevamente.**

El nombre de la aplicación está en el archivo *Manifest* (manifests > AndroidManifest.xml ) en el tag *application*, atributo *label.* Dentro del atributo está el valor string(@string/app\_name ) que lleva a *string.xml* donde encontramos **<string name=”app\_name”>AppPractica1</string>**.

Aquí es donde podemos cambiarlo por el que queramos.

Luego volvemos a correr la app, que al volver a ser compilada y lanzada muestra el cambio en el menú de aplicaciones.

**¿Y el ícono?**

Desde el propio IDE, podemos cliquear sobre *app,* el menú desplegable *New > Image Asset.* Aparece un cuadrito de *Configuration Image Asset* donde se puede configurar el *Foreground* (imagen/ícono/color) y el *Background* (fondo/color).

Una vez seleccionado el ícono tendremos que referenciar indicando cuál va ser el ícono que vamos a querer usar.

Vamos al archivo *res > mipmap > ic\_launcher,* ahi vemos que los antiguos íconos eran extensión *.webp* y los nuevos *.png* manteniendo el mismonombre.

Por eso borraremos los antiguos, con *.webp* para no desperdiciar espacios sin uso ni eficiencia en nuestro código.

Una vez que nos aseguramos que en el Manifest se toma bien nuestro ícono nuevo, *Manifest* (manifests > AndroidManifest.xml ) en el tag *application*, atributo *icon,* dentro está el valor (android icon=”@mipmap/ic\_launcher” ).

Corremos nuevamente la aplicación, y yendo al dashboard podremos ver el nuevo ícono.