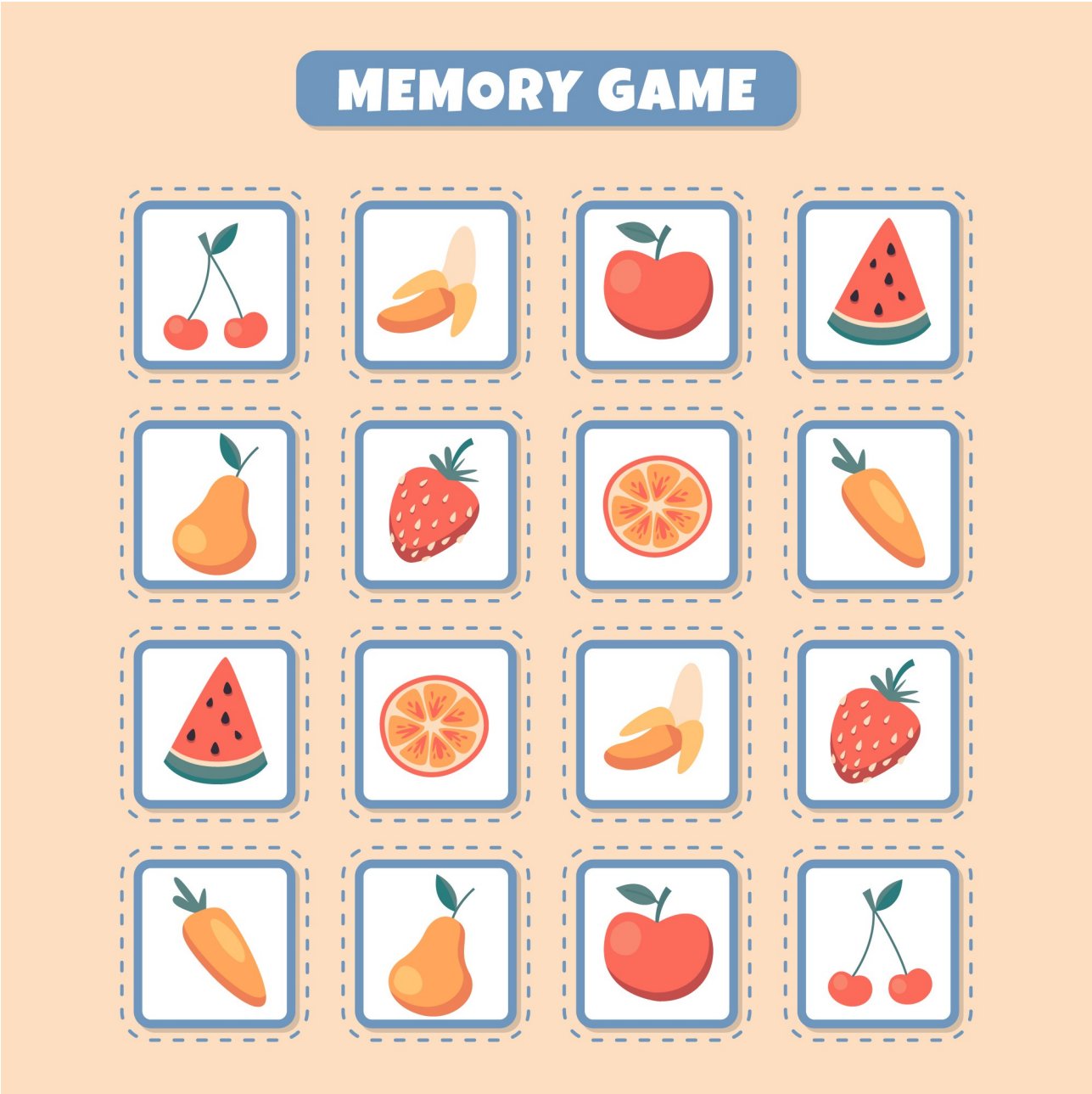


Documentación trabajo unidad 3



Índice

1.	Introducción.....	3
2.	Descripción del proyecto.....	3
3.	Requisitos.....	4
4.	Configuración del entorno.....	4
5.	Estructura del proyecto.....	5
6.	Instalación.....	7
7.	Uso.....	7
8.	Pruebas.....	10
9.	Contacto.....	10

Introducción

El trabajo de esta unidad está centrado en aplicar los conocimientos aprendidos de esta unidad para crear un programa. Conocimientos tales como manejar eventos, RecyclerView o Adaptadores por mencionar algunos.

Esta aplicación sirve para reunir todos estos conocimientos y demostrar principalmente el uso de adaptadores y adaptadores. Se trata de un juego de memoria, que aprovecha la estructura del GridLayout para colocar las cartas.

Descripción del Proyecto

Su principal objetivo es el entretenimiento, aunque también cuenta con otros objetivos menores derivados de la naturaleza del programa. Tales como la mejora de la memoria o de la concentración y con pequeñas modificaciones del programa puede tener también una finalidad educativa.

El programa cuenta con las siguientes características:

- El tablero que contiene las cartas. Es la principal funcionalidad del programa.
- Botón de nueva partida. En caso de que por alguna razón haya que empezar la partida de nuevo, y no se quiera esperar a encontrar todas las parejas.
- Botón de salida. Para salir directamente a la pantalla principal del móvil.
- Botón de cronómetro. Desactivado por defecto. En el momento en que se activa, comienza a correr el tiempo.

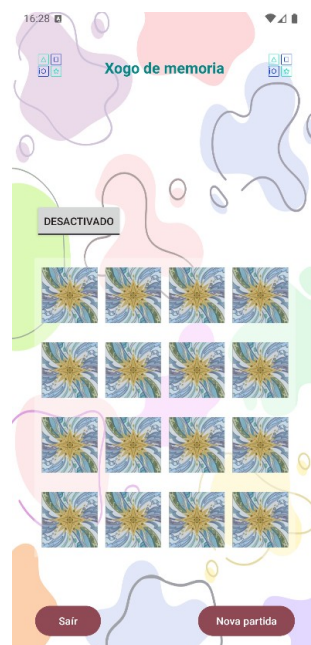


Figura 1: Interfaz de la aplicación

Partes a mejorar o añadir para una versión posterior.

- Ya que hay un cronómetro, se podría incorporar una lista de mejores puntuaciones en base al tiempo.
- Añadir un numero de vidas a la partida para añadir más emoción.
- Añadir animación a las cartas cuando se giran.

Requisitos

-Android Studio (version Koala 2024.1.2 o superior)

-JDK de Android (version Android API 35)

-SDK 11+

Configuración del Entorno

1. Clona el repositorio del proyecto desde: https://github.com/Iriaft21/Tarea_Unidad_3.git

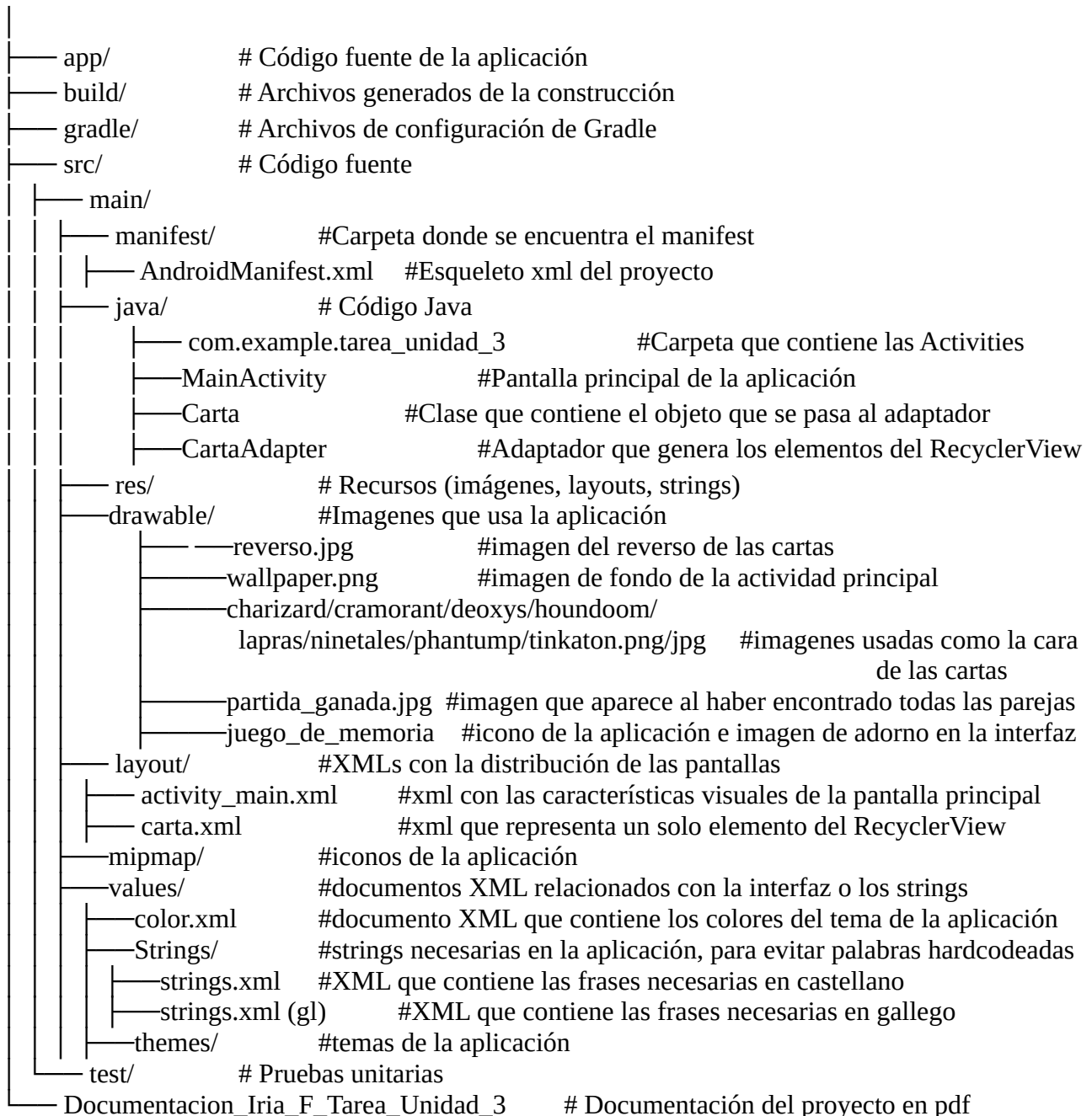
Selecciona la carpeta donde será clonado el proyecto y ejecuta el git clone desde el Git Bash.

2. Abre el proyecto en Android Studio.

3. Asegúrate de que el SDK de Android esté configurado correctamente.

Estructura del Proyecto

/Tarea_unidad_3



Relación de archivos más importantes:

- **MainActivity.** Contiene el funcionamiento principal de la aplicación. Entre los métodos más destacados están:
 - **generarNuevaPartida().** Método empleado tanto al pulsar el botón de nueva partida como al haber encontrado todas las parejas. En él se procede a poner a 0 tanto el cronómetro como el número de parejas hechas. Se vacía la lista donde se almacenan las cartas cada vez que se seleccionan dos cartas y se baraja la lista que contiene las cartas. Por último en un handler, se procede a llamar a otro método que pondrá todas las cartas boca abajo. El handler es necesario, ya que si no estuviera en el caso de encontrar todas las parejas no se mostraría la última que se seleccionó.
 - **onClick:** método que se activa al detectar un click en una de las cartas. Comprueba que la carta pulsada aún esté sin pareja, que se pueda voltear y que no se haya pulsado dos veces y la cuenta como pareja. Una vez hecha la comprobación, se espera a la segunda carta. Una vez tengamos las dos, se comprueba si son la misma carta o no. Dependiendo del resultado se mandan a un método u otro.
 - **aciertaPareja():** aquí es donde se va si ambas cartas resultan ser pareja. Lanza un aviso al usuario de que son pareja. Se aumenta el conteo de parejas encontradas, avisa al sistema de que son pareja y se vacía la lista que las contenía para albergar el siguiente par de cartas. También se comprueba si ya se han encontrado todas las parejas, si se ha hecho se mueve al método **finalizarJuego()**.
 - **FalloPareja():** aquí se va si las dos cartas seleccionadas son distintas. Lanza un aviso para el usuario, diciendo que ambas cartas son distintas. Luego mediante un handler se le da la vuelta a esas dos cartas y se vacía la lista que contenía las cartas.

El handler es necesario, ya que de no haberlo nos encontraríamos con el problema de que al seleccionar dos cartas distintas no nos enseñaría la última en seleccionar.
 - **FinalizarJuego():** aquí se entra en el caso de que se hayan encontrado todas las parejas. Hay un handler dentro de otro para que al encontrar la última pareja, haya un poco de tiempo antes de que salga la imagen de felicitación que desaparecerá unos segundos después. También llama a la función **generarPartidaNueva()** para crear una nueva partida en caso de que se quiera seguir jugando.
 - Además de eso en el método **onCreate** se crea y emplea un cronómetro para registrar el tiempo que toma en completar el juego en el caso que se desee utilizarlo. Así como los botones de Salir y de Nueva partida.
- **CartaAdapter:** adaptador del RecyclerView que convierte los elementos de clase Carta en una lista de vistas en la interfaz.
- **Carta:** clase que se usa para crear las cartas. Contiene una serie de atributos entre los que destaca las imágenes de la cara, del reverso, si una carta está registrada como emparejada o el valor de dicha carta.

Entre los elementos usados se encuentran dos elementos que no fueron dados en clase.

- Handlers: se emplea para asegurarnos de que el usuario pueda ver las cartas volteadas durante un tiempo determinado ya que sin él no se podría apreciar la segunda carta de una pareja o la última de las cartas en ser emparejada.

<https://es.stackoverflow.com/questions/262132/manejo-de-handler-y-runnable>

- Cronómetro: su uso está condicionado a una funcionalidad. Dicha funcionalidad es cronometrar cuánto se tarda en resolver el juego. Por tanto es necesario para el juego.

Url que se tomó de referencia para implementar el cronómetro:

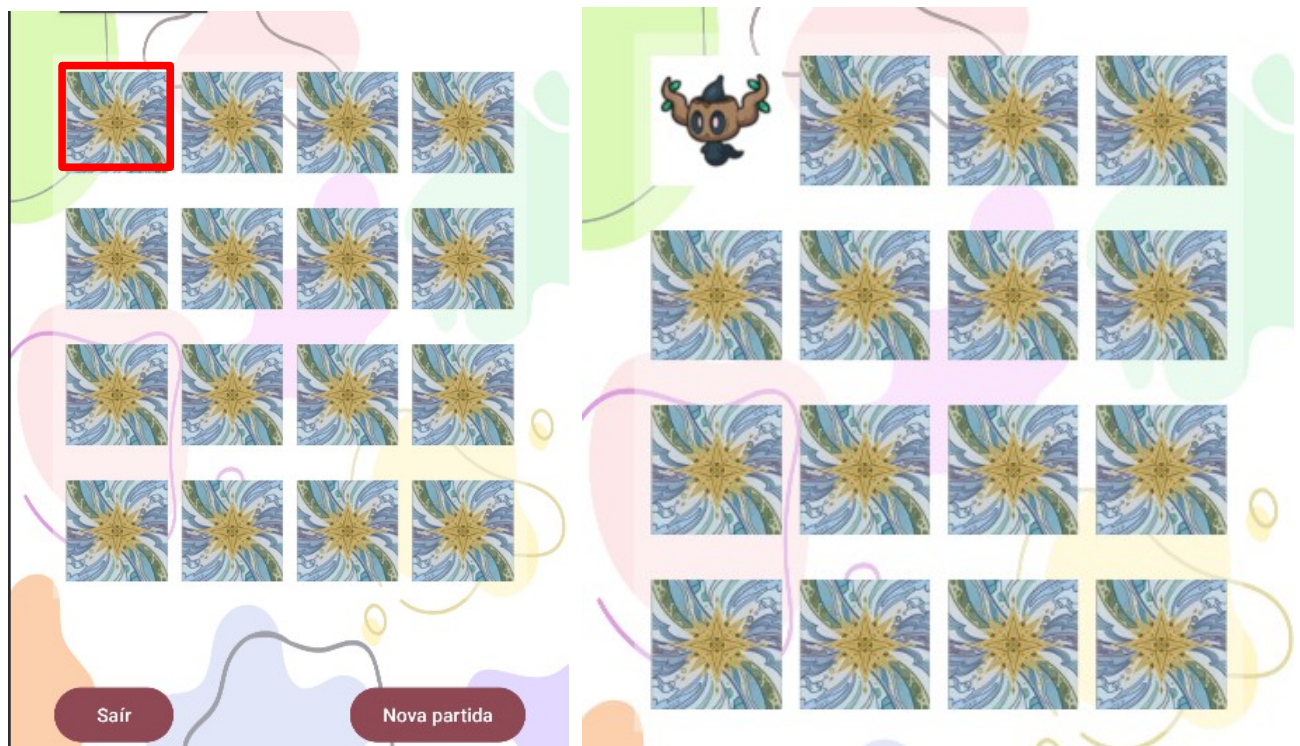
<https://abhiandroid.com/ui/chronometer#gsc.tab=0>

Instalación

1. Con Android Studio abierto, selecciona 'Run' > 'Run app'.
2. Conecta un dispositivo Android o utiliza un emulador.
3. Espera a que la aplicación se compile y se instale automáticamente.

Uso

1. Se selecciona una de las cartas para voltear.



2. Se selecciona la segunda carta y la aplicación avisará si son cartas iguales o diferentes.



Figura 2: Aviso que sale cuando las cartas son diferentes

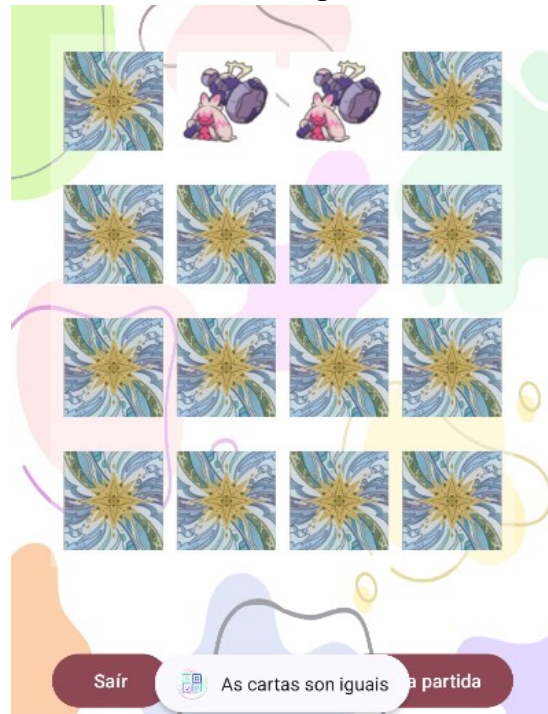


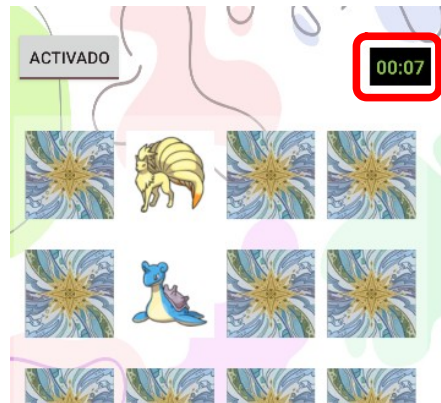
Figura 3: Aviso que sale cuando las cartas son iguales

3. Se siguen seleccionando cartas hasta que solo quede una pareja. Una vez encontrada la última pareja, aparece una felicitación que desaparece a los segundos y genera una nueva partida.



Figura 4: Mensaje al completar las parejas

- En caso de que se desee jugar cronometrando el resultado, pulsar el botón que pone desactivado. Eso habilitará el cronómetro. En caso de tenerlo activado al ganar una partida o generar una nueva, se pondrá a cero. Desactivarlo y volverlo a activar también lo pone a 0.



- Para salir de la aplicación basta con pulsar el botón de Salir/Saír.



- Para generar una nueva partida, pulsar el botón de Nueva partida/Nova partida.



Pruebas

Entre las pruebas que se han realizado se encuentran:

- Pulsar dos veces sobre la misma carta para ver si la cuenta como pareja.
- Activar y desactivar el cronómetro para ver si paraba la cuenta o lo ponía a cero.
- Pulsar en cartas ya emparejadas para ver si las añadía a las lista de cartas seleccionadas.
- Pulsar las cartas de forma muy rápida para comprobar que les de la vuelta correctamente.

Contacto

- Autor: Iria Fernández Trabazo
- Email: iriafr@gmail.com
- GitHub: <https://github.com/iriaft21>