

Práctica 2

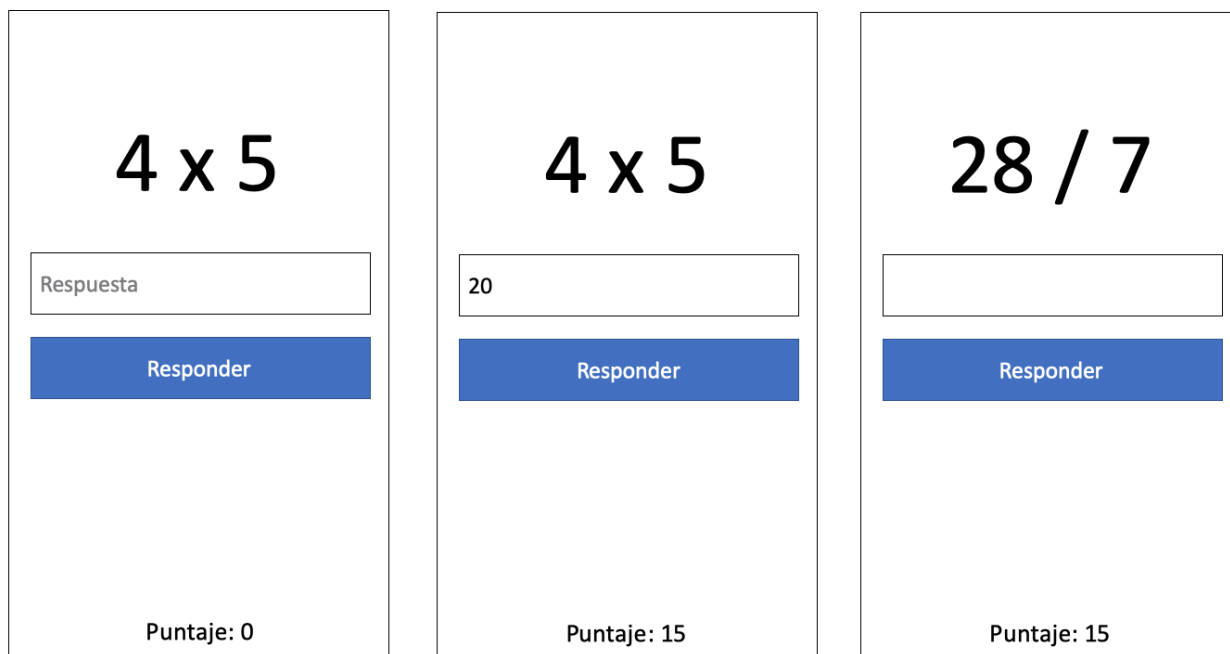
Objetivo

En esta práctica debe realizar una aplicación Android capaz de interactuar con un usuario. Para eso debe usar los listeners `OnClickListener` y `onTouchListener`.

Consigna

Icesi Games en asociación con Matileo lo contrata para que realice una aplicación que permita afiazar el conocimiento en aritmética de los estudiantes de Icesi.

Matileo decidió que la aplicación se llame Math Challenge y que consista en presentarle una serie de preguntas y que ofrezca un puntaje a partir de las respuestas.



The figure displays three sequential screenshots of the Math Challenge application interface. Each screen features a large math problem at the top, a text input field for the answer, a blue 'Responder' button, and a score display at the bottom.

- First Screenshot:** The problem is 4×5 . The input field is empty and labeled 'Respuesta'. The score at the bottom is 'Puntaje: 0'.
- Second Screenshot:** The problem is 4×5 . The input field contains the number '20'. The score at the bottom is 'Puntaje: 15'.
- Third Screenshot:** The problem is $28 / 7$. The input field is empty. The score at the bottom is 'Puntaje: 15'.

Figura 1. Pantalla de Math Challenge

Para las preguntas cree un banco de preguntas (arreglo de preguntas) de modo que vayan saliendo de forma aleatoria en la pantalla. Para eso cree una clase `Pregunta` en la que tenga la pregunta y la respuesta para que pueda hacer la comprobación directamente.

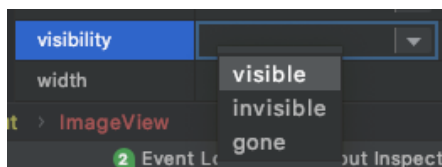
AVANZADO

Limite el juego para que dure 30 segundos. Una vez terminado el tiempo, puede volver a jugar si da click en el botón “Intentar de nuevo”.

Encuentre la forma de que el contador se muestre todo el tiempo en pantalla.

<div>28</div> <div>4 x 5</div> <div>Respuesta</div> <div>Responder</div> <div>Puntaje: 0</div>	<div>12</div> <div>12 x 16</div> <div>Respuesta</div> <div>Responder</div> <div>Puntaje: 15</div>	<div>0</div> <div>28 / 7</div> <div>Respuesta</div> <div>Responder</div> <div>Intentar de nuevo</div> <div>Puntaje: 15</div>
--	---	--

Note que el botón “Intentar de nuevo” aparece cuando temporizador finaliza. Usted puede hacer aparecer y desaparecer elementos visuales usando la propiedad visibility.



Esta propiedad puede ser VISIBLE, INVISIBLE (el botón aún permanece activo, solo que ahora se transparente) y GONE (es como si el botón ya no estuviera y no hay forma de darle click al botón).

En código también puede cambiar la visibilidad usando

```
miBoton.setVisibility(View.GONE)
```

Para ocultar el botón

```
miBoton.setVisibility(View.VISIBLE)
```

Para hacer aparecer el botón