

TP Videojuegos

Ejercicio 2

El objetivo de este ejercicio es usar el patrón **Observer** para eliminar las dependencias entre los componentes de la clase **GameManager** y la pelota (en el código que escribimos en la última clase).

Paso I:

Modifica la clase **GameCtrlInputComponent** para que no tenga llamadas a `gm->getBall()`. Para poder hacer esto, hay que mover el código que aparece en las líneas 20-24 a la clase **BallPhysics** y ejecutarlo cuando empieza una ronda.

IMPORTANTE: para poder acceder a la pelota desde todos los métodos de la clase **BallPhysics**, es decir desde el método donde has puesto el código de las líneas 20-24, lo más fácil sería poner la referencia en un atributo al principio del método `update` (`ball_=o`). Hay una solución mejor pero requiere añadir un método `init` a las interfaces de los componentes y invocar ese método cuando se instala un componente en un container.

Paso II:

Modifica la clase **GameManagerPhysics** para que no tenga llamadas a `gm->getBall()`.

Mover el código que aparece aparece en las líneas 23-26 y 32-35 a la clase **BallPhysics** y ejecutarlo cuando acaba una ronda.

Elimina las comparaciones que aparecen en las líneas 20 y 28. En lugar de esas comparaciones haz que la clase **BallPhysics** mande un mensaje adecuado cuando la pelota sale del lado derecho o izquierdo y que el **GameManager** incrementa el marcador cuando reciba dichos mensajes.

Ahora la clase **GameManagerPhysics** no tiene nada y puedes eliminarla.

Paso III:

En la clase **GameManager**: (1) elimina el parámetro `ball` del constructor; (2) elimina el atributo `ball_`; (3) elimina el método `getBall()`.