



## Explicación sobre las clases utilizadas en el diagrama

A continuación doy una explicación sobre como funciona a grandes rasgos las clases que implemente en mi diagrama UML para el ejercicio 1 de la practica 1A.

### Partida

En esta clase es la que se situaría el juego, donde se tiene **un mapa** el cual contiene tanto al jugador como al escuadrón enemigo, además esta clase tiene los métodos para empezar el juego, para terminarlo, en caso de haber ganado, y un game over para cuando pierda el jugador.

### Mapa

Esta clase es donde se sitúan el **jugador y el escuadrón de enemigos**, la hice en una clase aparte a la de partida porque si no cada vez que se quiera hacer algo en el mapa, estaría afectando a la partida cuando no es necesario y lo mismo para el caso opuesto, si quisiera comenzar la partida o decir que gano, estaría usando la clase mapa muchas veces, lo cual no está bien. Los métodos son:

- `borrarEnemigo()` y `borrarJugador()` sirven para que una vez su energía sea menor o igual a 0, se borren del mapa.

### Jugador

Esta clase es la que le da la oportunidad al jugador o usuario, de controlar **un tanque** para poder jugar al juego, ya que sin esta clase no se distinguiría un tanque del grupo de tanques enemigos del tanque que controla el jugador. Esta clase cuenta con los métodos de:

- `moverTanque()`: el cual mueve al tanque del jugador hacia la dirección que este quiera.
- `dispararTorreta()`: este método sirve para hacer un llamado a la torreta que tiene el tanque del jugador y así poder disparar en la dirección que el jugador quiera.

### Tanque

Con esta clase modelizo el comportamiento de lo que serían los tanques, estos cuentan con **una torreta**, su posición en X y en Y dentro del mapa, una variable la cual dice si el tanque se encuentra deshabilitado o no ya que al disparar demasiadas balas seguidas éste se deshabilita, y la energía del mismo esta sería como la vida del tanque, una vez llega a 0, se elimina el tanque del mapa. Los métodos que utilizo en esta clase son:

- `mermarEnergia()` el cual le va bajando la energía al tanque si recibe un disparo.
- `deshabilitar()`: este método pone en true la variable deshabilitado.
- `Habilitar()`: este método esta ya que si dejo de disparar el tanque por estar deshabilitado, al cabo de un rato se lo puede volver a habilitar.
- `mover()`: la función de este método es mover el tanque por el mapa.
- `getPosicionX()` y `getPosicionY()`: estos métodos son para obtener la posición del tanque en el mapa.

### Torreta

Esa clase es la que modeliza la torreta, la misma tiene como variables un boolean que dice si puede disparar o no, esta la uso para saber si esta caliente, y la variable `cantDisparos` que sirve para saber cuántos disparos seguidos está haciendo la torreta. Este tiene una relación de composición con la torreta, ya que sin el tanque la torreta no podría existir y su cardinalidad es de 1 porque cada tanque solo tiene una torreta. Los métodos que tiene esta clase son:

- `estaCaliente()`: me devuelve un boolean que me dice si esta o no caliente la torreta.
- `disparar()`: en base a la dirección que se pasa por parámetro, la torreta empieza a disparar.
- `bajarCantDisparos()`: este método sirve para que cuando no se este disparando con la torreta, la variable `cantDisparos` empiece a bajar y no se caliente.
- `calentar()`: una vez se haya pasado de un límite de balas disparadas seguidas, la torreta pasara a un estado de "caliente" donde no podrá disparar, es decir, `puedoDisparar` se pondrá en false.

## Escuadrón Enemigo

Esta clase modela al escuadrón enemigo, las variables que tiene son la dirección y un boolean para saber si encontró un adversario, en caso de haber encontrado uno en su ángulo de visión del radar, este se pondrá a disparar. El único método que tiene esta clase es el de mover() ya que el escuadrón se mueve en conjunto hacia una misma dirección.

## Radar

Esta clase sirve para modelar el radar que contiene la torreta, este cuenta con una sola variable la cual guarda el ángulo de visión del mismo. El único método que tiene es hayEnemigo() el cual devuelve verdadero si se encuentra un enemigo en el ángulo de visión. Lleva una relación de composición porque el radar no puede existir sin la torreta en el juego.

## Bala

Esta clase modela la bala que dispara la torreta, esta cuenta con su dirección de disparo y su calibre. Esta tiene una relación de agregación con la torreta ya que si puede existir sin la misma, porque por ejemplo puede ser que un tanque enemigo haya disparado una bala y si el jugador mata a este tanque la bala que disparo seguirá en el mapa por más que el tanque enemigo haya sido eliminado. Los métodos que tiene son:

- colision(): sirve para saber cuando un bala choca con un tanque.
- generarBala(): este método genera la bala en el mapa.