

# Informe Tp Pokemones

Integrantes: Vucetic Ivo.

## **Modalidad del torneo:**

El torneo desarrollado cuenta con un cupo de **8 entrenadores**. Para los que cada entrenador puede usar **3 pokemones** durante el transcurso del torneo, además de contar con un conjunto de cartas de tres tipos diferentes (luego se comentará esta implementación), tienen dos cartas de cada tipo y cada entrenador podrá usar solo tres durante el torneo.

El torneo cuenta con 3 rondas. Por cada ronda habrá diferentes batallas entre dos entrenadores en donde el entrenador perdedor quedará eliminado del torneo y el ganador pasará de ronda. Se disputará de esta manera hasta que resulte el ganador del torneo.

En cada ronda, un entrenador podrá presentar uno de los pokemones que tiene para poder enfrentarse a duelo con el pokemon del entrenador rival.

Cada entrenador tiene un Nombre, una lista de pokemones, una lista de cartas hechizo(luego se comentará), puntos de batalla y una categoría.

Cada pokemon tiene un nombre, puntos de experiencia, un nivel de escudo, nivel de vitalidad, un nivel de fuerza de ataque, una categoría y los puntos máximos a los cuales podrían llegar dependiendo de su categoría.

Tanto el entrenador como sus pokemones durante el transcurso del torneo podrán subir de categoría(estó realmente está pensado para un número mayor de rondas y batallas) entre las que se encuentra **Principiante, intermedio y avanzado**.

La categoría de un pokemon está dada por su nivel de experiencia. A medida que el pokemon sube, el máximo al cual podría llegar cada uno de sus atributos lo hace también, esto quiere decir que un pokemon principiante

que alcance un máximo en fuerza estará por debajo de un pokemon que alcance el máximo del mismo atributo que sea de tipo Intermedio o avanzado. Esto genera la diferencia entre las categorías de los pokemones generando que haya inevitablemente pokemones más fuertes que otros.

La categoría de un entrenador está dada por la suma del nivel de experiencia de cada uno de sus pokemones. A medida que el entrenador pasa de categoría se le sumará una cantidad de puntos de batalla, los cuales luego los podrá cambiar por atributos para sus pokemones.

El torneo cuenta con tres arenas, Este, Central y Oeste. En dichas arenas se disputarán los enfrentamientos de cada una de las rondas.

La arena puede pasar por diferentes estados que pueden ser: **Preliminar, Batalla y Definición.**

**Preliminar:** Se presentan los entrenadores y los pokemones correspondientes.

**Batalla:** Durante este periodo se llevarán a cabo las batallas en si misma, lanzamientos de golpes y hechizos.

**Definición:** Se define el ganador y lo que podrá obtener como recompensa por haber ganado la batalla.

Una vez en la arena, de forma aleatoria se define quien comienza a atacar, y luego el otro realiza el contrataque. Cada ataque consta de tres golpes consecutivos, un golpe inicial que será el mismo para todos los pokemones, la recarga(solo algunos cuentan con dicha posibilidad) y un golpe final.

Existen dos clases bases de pokemones que serán **común y legendario**. Además hay clases que van a poder mejorar estas, como **Fuego, Agua, Hielo, Hada y Roca**.

En el siguiente cuadro se muestra los atributos que tendrán las clases.

TIPO	vitalidad	escudo	fuerza
Común	100	20	25
Legendario	200	40	50
Agua	+500	+100	+120
Hielo	+500	+120	+100

Fuego	+480	+100	+140
Roca	+520	+130	+70
Hada	+500	+60	+160

En el siguiente cuadro se muestra el modo de atacar y de recibir daño.

Tipo	Golpe inicial	Recarga	Golpe final	Recibe daño
Común	Todos los tipos realizan el mismo ataque inicial, infligiendo al adversario un daño igual a su fuerza de ataque, luego su fuerza de ataque se reduce a la mitad	Incrementa un 10% la fuerza y la vitalidad.	Provoca un daño igual a la fuerza que le ha quedado.	El escudo recibe todo el daño, solo se reduce la vitalidad cuando este se agote.
Legendario		Si posee la capacidad de <b>recarga legendaria</b> su fuerza aumenta un 10%	Provoca un daño adicional del 50% de su escudo.	El escudo recibe todo el daño, solo se reduce la vitalidad cuando este se agote y su fuerza aumenta un 5%.
Fuego		Si posee la capacidad de <b>recarga de fuego</b> Incrementa un 10% la fuerza y la vitalidad. Sino no será capaz de recargar.	Provoca al adversario un daño igual a su fuerza mas un 25% y luego la fuerza se agota por completo (queda en cero)	Si dispone de <b>aura de fuego</b> si vitalidad sube un 10%.
Agua		Si posee la capacidad de <b>recarga de agua</b> Incrementa un 10% la fuerza y la vitalidad. Sino no será capaz de recargar.	El escudo absorbe todo el daño, solo cuando éste se agota comienza a decrementarse la vitalidad	Si dispone de <b>aura de agua</b> su escudo sube un 10%. Luego El escudo absorbe todo el daño, solo cuando éste se agota comienza a decrementarse la vitalidad
Hielo		Si el Pokemon posee capacidad de "gran	Provoca al adversario un daño igual a su	Si dispone de aura de hielo su

		recarga" su fuerza tomará un valor de 400 pero no recargará su vitalidad, si no posee dicha capacidad, tendrá el comportamiento habitual.	fuerza menos un 10% pero conserva la misma fuerza	escudo y vitalidad suben un 5% luego El escudo absorbe el 75% del daño y la vitalidad el otro 25%
Roca		No dispone de recarga	Provoca a su adversario un daño igual a su fuerza mas un 20%, no se agota su fuerza.	Si dispone de <b>aura de roca</b> su vitalidad y escudo aumenta un 10% , del daño que recibe un 80% es para el escudo y una vez que se agota reduce el 20% de la vit
Hada		No dispone de recarga	Provoca a su adversario un daño igual a su fuerza mas un 15% y su escudo aumenta 30 puntos.	Si posee aura de hada su escudo aumenta un 5%. Primero se reduce el escudo y luego la vitalidad.

Una vez que termina la batalla, el entrenador que gana es aquel en que la suma de los atributos finales (vitalidad, escudo y fuerza) del pokemon es mayor.

Al ganar una batalla el pokemon gana 3 puntos de experiencia, y el perdedor solo obtendrá un punto. El entrenador que gana obtendrá 200 punto de batalla y podrá quedarse con un pokemon aleatorio de los pokemones del perdedor.

El entrenador cuenta con dos cartas de cada tipo. Las cartas hechizo son: **niebla, viento y tormenta** que depende del tipo de pokemon el daño que le causen a sus atributos. Pero durante el transcurso del torneo solo podrán usar 3 cartas del total.

Tanto los entrenadores como los pokemones que son comunes se podrán clonar.

Se implementaron tres ventanas para poder trabajar con el torneo.

### **1)Ventana Menú**

Es la de bienvenida la cual cuenta con un botón para ver de qué se trata el torneo y con un botón de inicio en el caso de que se quiera participar.

### **2)Ventana Inscripción**

En esta pantalla vamos a poder poner los datos tanto del entrenador como de los pokemones que tiene.

#### ***Partes de la ventana:***

-**Nombre entrenador:** Se debe poner el nombre del entrenador que quiera participar, sin ninguna restricción, se puede poner el nombre que se quiera.

Una vez que se ponga el parámetro anteriormente descrito con el botón Agregar se realiza el alta del entrenador y se habilita la posibilidad de inscribir los pokemones que usará para el torneo (los cuales serán 3).

-**Nombre pokemon:** Se debe poner el nombre del pokemon que se quiera registrar, sin ninguna restricción, se puede poner el nombre que se quiera.

-**Tipo:** Aquí solo tenemos dos posibilidades como anteriormente se registró, *común o legendario*.

- **Tipo Elemento:** Se podrán registrar pokemones de *agua, fuego, hielo, roca y hada*. En el caso de que el pokemon a registrar no cuente con la particularidad de algún elemento, se podrá dejar vacía esta opción y de igual manera se registrará el pokemon.

Una vez que se completen los parámetros necesarios para el registro se habilitará el botón agregar para que se pueda dar el alta del pokemon.

Finalizado el registro de los entrenadores con sus respectivos pokemones se habilitará el botón Inicio Torneo para poder comenzar con el mismo.

### 3) Ventana Arenas

En esta pantalla se visualizará las batallas correspondientes a cada ronda. Cuenta con tres paneles en que en cada uno se mostrará cada arena, con los enfrentamientos que se darán en cada una de ellas y un panel inferior el cual muestra el paso de cada ronda, finalización de la misma y el ganador del torneo.

#### Persistencia del Torneo

En la clase Main del programa está la posibilidad de iniciar la ejecución para poder ingresar los datos de forma manual con el ingreso de cada entrenador con sus pokemones, esto se puede hacer poniendo **true** en la variable booleana `escribirInscripcion`. En el caso de que se ponga **false**, hay un archivo llamada `Inscripcion.bin` el cual ya tiene cargado un listado con los entrenadores y sus respectivos pokemones de esta forma al inicializar el programa podremos iniciar el torneo desde el botón `IniciaTorneo` y se ejecutara.

El tipo de persistencia que se realiza en el trabajo es binaria.

Al finalizar cada ronda, se persistirá la información del listado de los entrenadores ganadores, esto quedará guardado en un archivo `.bin`, dependiendo de en qué ronda se encuentren quedarán en `OctavosDeFinal.bin`, `CuartosDeFinal.bin` y `SemiFinal.bin`.

Al iniciar la jornada se tomará la información dependiendo de la ronda de los archivos descriptos anteriormente.

#### Patrones de diseño:

Durante el transcurso del código se implementó:

**Patrón singleton** para realizar solo una instancia del torneo.

**Patrón factory** para la creación de los pokemones. El

**Patrón decorador** se implementó para sumar funcionalidad en los pokemones base (común y legendario).

**Patrón template** se utilizó para realizar la forma de ataque del pokemon.

**Patrón MVC:** Para realizar esta implementación se decidió implementar solo un controlador, que se encargue de manejar al modelo y a las correspondientes vistas que se desarrollaron. El controlador tendrá la posibilidad de “conocer” al modelo y a la vista. La vista no tiene relación con el modelo, ni el modelo con la vista, sino que el controlador será el intermediario de ambos. Aquí es cuando se utilizó el **Patrón Observer\Observable** mediante el cual el modelo le informa al controlador de los cambios que tuvo y luego el controlador le informa a la vista para poder mostrarlo por la interface correspondiente.

El **patrón State** se utilizó para darle un orden en las acciones que se desarrollan en las arenas.

El programa se entrega sin errores y con el javadoc correspondiente.