

MANUAL TECNICO DE POKEESMERALDA

by Juan Pablo Orizabal Gil

Carné: 201730318

MANUAL TÉCNICO

El software se puede utilizar en cualquier sistema operativo gracias a java, ademas se puede correr en la mayorita de computadoras que tengan instalado java 6 o superior. Se utilizo el JDK8 de java, y el IDE Netbeans.

El software es un juego que permite la creación de mascotas llamadas pokemon; podemos interactuar con ellas, tienen un nombre, una imagen en código ascii, así como podemos cuidarlas. Se tiene la opcion de darle de comer, hacer alguna actividad con él. También se tiene la posibilidad de combatir con otros entrenadores pokemon, esto se puede pero con otro software, cada jugador debe de tener su software abierto, o mas que el software tener un programa independiente ejecutándose. Se tiene la posibilidad de ver distintos reportes de los pokemon, entre los mas destacados ver quien es el mejor y peor pokemon guerrero, y el de ver quien ha consumido mas alimentos saludables y cual menos.

El software se divide en 4 clases: Actividades, Comida, Pokemon y PokeEsmeralda(clase main).

Clase Actividades

Se tienen diferentes atributos(privados): nombre de la actividad, tipo de actividad, factor que hace de salud la actividad. Existe un constructor para facilitar la instanciación de la clase.

Tiene los métodos de sets y gets de cada atributo, lo cual nos sirve para poder manipular el valor de los atributos mediante acciones.

Clase Comidas

Se tienen diferentes atributos(privados): nombre de la comida, tipo de comida, factor que hace de salud la comida. Existe un constructor para facilitar la instanciación de la clase.

Tiene los métodos de sets y gets de cada atributo, lo cual nos sirve para poder manipular el valor de los atributos mediante acciones.

Clase Pokemon

La clase pokemon es de las que mas cosas ocupa, tiene muchos atributos, los cuales son: nivel, vida, experiencia, nombre, todos estos del pokemon; así como también la cantidad de pokemon derrotados, su estado de pelea y su estado de vida, su imagen ascci, la cantidad de veces que ha huido y desmayado, por ultimo se tiene dos instancias: una de la clase Comida y uno de Actividades(estos sirven para llevar el control de lo que han hecho con estas dos clases). Todas de visibilidad privada.

Existen métodos:

Existen los sets y gets, que sirven para manipular el valor de los atributos mediante acciones.

Método atacar: este método lo que hace es crear un numero aleatorio basándose en el nivel del pokemon para luego mediante ese numero generar el código que se enviara al método generar código, devuelve un valor entero el cual es el daño que esta haciendo el pokemon en cuestión.

Método huir: lo que hace es cambiar el estado de pelea del pokemon a “huyo” y regresar una letra R que sera lo que usara el método generar código para imprimir en pantalla al enemigo, por eso es de tipo String.

Método cambiarEstadoPokemonPelea: este método lo único que hace es cambiar el estado del pokemon en pelea, devuelve un valor String que servirá en la clase PokeEsmeralda.

Método recibirDaño: este método lo que hace es recibir el daño del pokemon enemigo mediante un parámetro, este valor lo usara para restarle vida al pokemon que este activo y si la salud del pokemon llega a menos del 25% este se desmayara y no peleara mas, incluso puede que muera el pokemon.

Método comer: este método recibe tres parámetros, el nombre de la comida, el tipo de comida y el factor de salud que hace, lo que hará es dependiendo del tipo de comida y del factor de salud que hace sumarle o restarle vida al pokemon, al igual que en el método recibir daño si la vida del pokemon baja del 25% se desmaya y si llega a 0 se muere. Lleva el control de comidas también, por medio de una variable.

Método actividad: este método recibe tres parámetros, el nombre de la actividad, el tipo de actividad y el factor de salud que hace al pokemon. Dependiendo del tipo de actividad este se le sumara o restara vida al pokemon, si la salud del pokemon disminuye del 25% se desmaya y si llega a 0 se muere, cambiándole el estado del pokemon respectivo. Lleva el control de actividades realizadas por medio de una variable que se va aumentando cada vez que se ejecuta.

Método recibirExperiencia: Este método lo único que hará es mediante un parámetro sumarle experiencia al pokemon en cuestión, y llama al método subir de nivel que mas adelante se explicara.

Método subirDeNivel: este método muestra en pantalla si y solo si el pokemon subió de nivel al recibir experiencia, cada vez que se llama verifica la experiencia y hace un calculo, y si el nivel que salio del calculo es mayor al nivel del pokemon automáticamente sobrescribiera el nivel del pokemon por el del calculo.

Método escogerImagenASCCI: tiene una característica especial, y es que es estático, es decir, se puede llamar sin necesidad de instanciar ya que se necesita saber la imagen que tendrá el pokemon a la hora de crearlo, muestra 5 diferentes opciones que se podrán utilizar, se pueden repetir imágenes.

Método guardarHistorialComidas: Este método va guardando el nombre, tipo y factor que hace la comida en un vector de tipo Comida, este nos servirá para llevar el control de las comidas del pokemon. Y aumenta la variable cuentaComidas en 1 cada vez que se guarda una comida.

Método mostrarHistorialComidasDescendentes: este método recorre el arreglo de comida desde 0 hasta la cantidad de cuentaComidas, mostrando el nombre, tipo y factor de salud de la comida en la posición en cuestión. Desde la primera hasta la última comida ingerida.

Método mostrarHistorialComidasAscendentes: es lo contrario al método mostrarHistorialComidasDescendentes. Muestra las comidas ingeridas por el pokemon desde la última hasta la primera.

Método guardarHistorialActividades: pide de parámetros los 3 atributos de actividades, con los cuales irá guardando y registrando cada actividad que se realice mediante el arreglo que se creó al inicio, y se movilizara mediante la variable cuentaActividades, la cual se incrementará cada vez que se guarda una actividad.

Método mostrarHistorialActividadesDescendentes: Recorre todo el arreglo de historialActividades que se creó al principio desde 0 hasta la cantidad que almacena cuentaActividades, mostrando en cada ciclo lo que hizo con esa actividad. Muestra desde la primera hasta la última actividad realizada.

Método mostrarHistorialActividadesAscendentes: Recorre todo el arreglo de historialActividades que se creó al principio desde la cantidad de cuentaActividades hasta 0, decremantandose cada vez que hace un ciclo, muestra desde la última actividad realizada hasta la primera.

Método contarComidasBuenas: este método cuenta la cantidad de comidas saludables que realizó el pokemon para ver quien es el que más saludable consumió.

Método contarComidaMala: este método cuenta la cantidad de comidas no saludables que realizó el pokemon para ver quien es el que menos saludable consumió.

Método mostrarEstadisticasPokemon: como su nombre lo indica, muestra las estadísticas del pokemon, su nombre, su nivel, su experiencia, entre otros.

Clase PokeEsmeralda

Esta clase es la que contiene al main, es decir, donde va a empezar a leer el programa. Existen muchos atributos que se utilizaron para poder llenar la logica que se propuso. Esta la opcionMenuPrincipal que sirve para moverse por los difentes menus del juego, se creo un vector de tipo Pokemon para el almacenamiento de los pokemon, lleva el historial de batallas, si hay pokemon vivos y listos para luchar, y claves que sirven para salir de ciertos ciclos en momentos muy especificos.

Método crearPokemon: En este metodo es donde se crean los pokemon, si ya existen 5 pokemon no dejara que se creen mas, y tambien si se quiere crear un pokemon con un nombre repetido marcara que no se puede poner ese nombre. Y por ultimo cuando se cumplan los requerimientos del pokemon pedira la imagen que desea e instanciara al pokemon dandole los atributos pertinentes.

Método escogerPokemonActivo: es de los mas importantes debido a que es el que maneja a que pokemon se le realizaran las acciones, si no existen pokemon que se puedan poner como activo no dejara entrar al menu.

Método revivirOEliminarPokemon: Aqui se podra eliminar o revivir a un pokemon, si no hay pokemon muertos no dejara entrar al menu.

Método eliminarPokemon: Aqui es donde desenvoca si se desea eliminar un pokemon, lo pone nulo.

Método actividades: muestra las actividades que existen y las divide entre las saludables y las no saludables, dependiendo de la que se desee se llamara al metodo de la clase pokemon para que realice la actividad, se le enviara los parametros correspondientes.

Método alimentacion: Es muy parecido al de actividades, pero este no dejara entrar si el pokemon activo esta muerto, debido a que solo con una actividad se puede revivir. Luego muestra las comidas que existen y enviara datos que sirvan como parametros al metodo comer de la clase pokemon.

Método menuBatalla: solo muestra que se esta en una batallas.

Método elegirTurno: Elige el turno del entrenador que empezara, mediante un numero aleatorio.

Método verificarSiHayPokemonParaLuchar: Es un metodo booleano que devolvera si hay pokemon para luchar.

Método verificarSiHayPokemonVivos: Es muy parecido al metodo anterior, solo que devolvera si hay pokemon vivos.

Método verificarSiHayPokemonMuertos: Es muy parecido al metodo anterior, solo que devolvera si hay pokemon muertos.

Método opcionesDePelea: Este metodo es el encargado de toda la batalla, con un ciclo mientras se podra seguir peleando hasta que una de sus condiciones no se cumpla, como que ya no hay pokemon vivos o disponibles para luchar, y llamara a otros metodos como leerCodigo, generarCodigo, repartirExperiencia, entre otros.

Método generarCodigo: con parametro un entero, es el que se llamara cuando se desea atacar, se envia el nombre seguido de “|” y el daño hecho.

Método generarCodigo: con parametro un String, que servira si se desea huir un pokemon enviando el nombre seguido de “|” y la letra R, la cual indica que un pokemon huyo.

Método verificarQuienGanoLaBatalla: Sirve para ver quien gano la batalla si el jugador o el contrincante, si se gano la batalla llamara al metodo repartirExperiencia, sino saldra de la batalla.

Método regresarPokemonASuEstadoPokemon: Este metodo sirve cuando un pokemon estaba desmayado en pelea, quiere decir que sigue vivo, si este es el caso solo mostrara que esta vivo, en lugar de que esta desmayado.

Método leerCodigo: Este entra en accion cuando quiere recibir el codigo del contrario. Dependiendo del codigo leera el nombre del pokemon enemigo y lo que esta haciendo, primero tratara de transformar el lado derecho del codigo a un entero, si lo logra quiere decir que ese es el daño que se esta haciendo, por lo cual llamara al metodo recibirDaño de la clase pokemon y dara como parametro el daño hecho, si no se puede convertir en entero dependera de que esta en ese lado del codigo, si es una R se le añadira al pokemon activo un desmayo del enemigo, si es una T quiere decir que termino la batalla, lo cual hará que se llame al metodo verificarQuienGanoLaBatalla.

Método repartirExperiencia: Pedira que se indique la cantidad de vida que se le bajo a los pokemon enemigos y luego dependiendo de cuantos pokemon vivos y muertos queden repartira la experiencia de una u otra forma, siempre dandole mas a los pokemon vivos que a los muertos o desmayados.

Método mostrarHistorialDeBatallas: solo muestra cuantas batallas ganadas, perdidas, y cuantas en general se han realizado.

Método reportes: Este metodo muestra un menu en el cual estan todos los reportes a los que se pueden acceder.

Método reporteEstadoPokemon: Este metodo mostrara el nombre, el estado y la imagen de cada pokemon, al terminar un pokemon tendrá la opcion de ver el siguiente o salirse.

Método reporteAlimento1: Muestra el historial de comida de cada pokemon en orden descendente, del primer alimento al ultimo que consumio. Al terminar un pokemon tendrá la opcion de ver el siguiente o salirse.

Método reporteAlimento2: Muestra el historial de comida de cada pokemon en orden ascendente, del ultimo al primer alimento que consumio. Al terminar un pokemon tendrá la opcion de ver el siguiente o salirse.

Método reporteVisual: Muestra la imagen de cada pokemon. Al terminar un pokemon tendrá la opcion de ver el siguiente o salirse.

Método reporteActividades: Muestra las actividades realizadas de cada pokemon en orden descendente, de la primera a la ultima actividad realizada. Al terminar un pokemon tendrá la opcion de ver el siguiente o salirse.

Método reporteMejorYPEorAlimento: Primero ordena mediante unas variables y ciclos quien fue el que mas alimentos saludables consumo y quien el que menos, luego muestra en pantalla el historial de comidas en orden descendentes de esos dos pokemon en cuestion.

Método reporteBatallas: Muestra las batallas realizadas así como las ganadas y perdidas.

Método reporteMejorPeorPokemonGuerrero: Usa ciclos y variables para ordenar quien fue el pokemon que mas pokemon enemigos desmayo, el que mas veces huyo de batalla, y el que mas veces se desmayo. Al tener la casilla del pokemon en cuestion, muestra en pantalla sus estadísticas.

Método mostrarMenuPrincipal: Muestra todas las opciones del menu principal.

Método main: es el metodo principal en el cual estan todas las opciones que se pueden elegir para navegar por el juego.